

*Public*

# Common Criteria Information Technology Security Evaluation

---

## STRONGV3P00\_In04Ipe of S5E9935 with Specific IC Dedicated Software

Version 0.6  
5<sup>th</sup> December 2023

### ST (Security Target) Lite

SAMSUNG ELECTRONICS RESERVES THE RIGHT TO CHANGE PRODUCTS, INFORMATION AND SPECIFICATIONS WITHOUT NOTICE.

Products and specifications discussed herein are for reference purposes only. All information discussed herein is provided on an "AS IS" basis, without warranties of any kind.

This document and all information discussed herein remain the sole and exclusive property of Samsung Electronics. No license of any patent, copyright, mask work, trademark or any other intellectual property right is granted by one party to the other party under this document, by implication, estoppel or otherwise.

Samsung products are not intended for use in life support, critical care, medical, safety equipment, or similar applications where product failure could result in loss of life or personal or physical harm, or any military or defense application, or any governmental procurement to which special terms or provisions may apply.

For updates or additional information about Samsung products, contact your nearest Samsung office.

All brand names, trademarks and registered trademarks belong to their respective owners.

© 2022 Samsung Electronics Co., Ltd. All rights reserved.

# Important Notice

Samsung Electronics Co. Ltd. ("Samsung") reserves the right to make changes to the information in this publication at any time without prior notice. All information provided is for reference purpose only. Samsung assumes no responsibility for possible errors or omissions, or for any consequences resulting from the use of the information contained herein.

This publication on its own does not convey any license, either express or implied, relating to any Samsung and/or third-party products, under the intellectual property rights of Samsung and/or any third parties.

Samsung makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does Samsung assume any liability arising out of the application or use of any product or circuit and specifically disclaims any and all liability, including without limitation any consequential or incidental damages.

Customers are responsible for their own products and applications. "Typical" parameters can and do vary in different applications. All operating parameters, including "Typicals" must be validated for each customer application by the customer's technical experts.

Samsung products are not designed, intended, or authorized for use in applications intended to support or sustain life, or for any other application in which the failure of the Samsung product could reasonably be expected to create a situation where personal injury or death may occur. Customers acknowledge and agree that they are solely responsible to meet all other legal and regulatory requirements regarding their applications using Samsung products notwithstanding any information provided in this publication. Customer shall

indemnify and hold Samsung and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, expenses, and reasonable attorney fees arising out of, either directly or indirectly, any claim (including but not limited to personal injury or death) that may be associated with such unintended, unauthorized and/or illegal use.

**WARNING** No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electric or mechanical, by photocopying, recording, or otherwise, without the prior written consent of Samsung. This publication is intended for use by designated recipients only. This publication contains confidential information (including trade secrets) of Samsung protected by Competition Law, Trade Secrets Protection Act and other related laws, and therefore may not be, in part or in whole, directly or indirectly publicized, distributed, photocopied or used (including in a posting on the Internet where unspecified access is possible) by any unauthorized third party. Samsung reserves its right to take any and all measures both in equity and law available to it and claim full damages against any party that misappropriates Samsung's trade secrets and/or confidential information.

**警告** 本文件仅向经韩国三星电子株式会社授权的人员提供，其内容含有商业秘密保护相关法规规定并受其保护的三星电子株式会社商业秘密，任何直接或间接非法向第三人披露、传播、复制或允许第三人使用该文件全部或部分内容的行为（包括在互联网等公开媒介刊登该商业秘密而可能导致不特定第三人获取相关信息的行为）皆为法律严格禁止。此等违法行为一经发现，三星电子株式会社有权根据相关法规对其采取法律措施，包括但不限于提出损害赔偿请求。

**Copyright © 2022 Samsung Electronics Co., Ltd.**

Samsung Electronics Co., Ltd.  
DSR, 1-1 SamsungJeonja-Ro, Hwaseong-Si, Gyeonggi-Do, Republic of Korea

Home Page: <http://www.samsungsemi.com>

# Table of Contents

<b>1 ST INTRODUCTION .....</b>	<b>7</b>
1.1 Security Target and TOE Reference.....	8
1.2 TOE Overview and TOE Description .....	9
1.2.1 Introduction .....	9
1.2.2 TOE Definition .....	9
1.2.3 TOE Features .....	17
1.2.4 TOE Life cycle .....	19
1.3 Interfaces of the TOE.....	22
1.4 TOE Intended Usage .....	23
<b>2 CONFORMANCE CLAIMS .....</b>	<b>24</b>
2.1 CC Conformance Claim .....	25
2.2 PP Claim .....	25
2.3 Package Claim .....	25
2.4 Conformance Claim Rationale .....	25
<b>3 SECURITY PROBLEM DEFINITION .....</b>	<b>27</b>
3.1 Description of Assets .....	28
3.2 Threats .....	30
3.2.1 Standard Threats .....	32
3.2.2 Threats related to security services.....	34
3.2.3 Threats related to additional TOE Specific Functionality.....	35
3.2.4 Threats related to Authentication of the Security IC.....	35
3.2.5 Threats related to External Memory .....	35
3.3 Assumptions.....	38
3.4 Organizational Security Policies .....	38
<b>4 SECURITY OBJECTIVES .....</b>	<b>42</b>
4.1 Security Objectives for the TOE.....	43
4.1.1 Standard Security Objectives .....	44
4.1.2 Security Objectives related to Specific Functionality (referring to SG4).....	46
4.1.3 Security Objectives for Added Function .....	47
4.2 Security Objectives for the Security IC Embedded Software .....	51
4.2.1 Clarification of “Treatment of User Data of the Composite TOE(OE.Resp-Appl)” .....	51
4.3 Security Objectives for the Operational Environment.....	52
4.3.1 Clarification of “Protection during Composite Product Manufacturing (OE.Process-Sec-IC)” .....	52
4.4 Security Objectives Rationale .....	53
<b>5 EXTENDED COMPONENTS DEFINITION .....</b>	<b>57</b>
5.1 Definition of the Family FCS_RNG .....	58
5.2 Definition of the Family FMT_LIM .....	59
5.3 Definition of the Family FAU_SAS .....	61
5.4 Definition of the Family FDP_SDC.....	62
5.5 Definition of the Family FIA_API .....	63

5.6 Definition of the Family FDP_URC .....	64
5.7 Definition of the Family FDP_IRA .....	65

## **6 IT SECURITY REQUIREMENTS ..... 67**

6.1 Security Functional Requirements for the TOE .....	68
6.1.1 Malfunctions.....	68
6.1.2 Abuse of Functionality .....	69
6.1.3 Physical Manipulation and Probing .....	70
6.1.4 Leakage .....	71
6.1.5 Random Numbers (DTRNG/DRBG).....	72
6.1.6 Memory Access Control .....	74
6.1.7 Cryptographic Support.....	76
6.1.8 Triple-DES Operation .....	77
6.1.9 AES Operation.....	77
6.1.10 Key Manager (KDF) Operation.....	77
6.1.11 Secure Hash Algorithm (SHA).....	78
6.1.12 Hash-based Message Authentication Code (HMAC).....	79
6.1.13 Rivest-Shamir-Adleman (RSA) Operation (optional).....	79
6.1.14 Rivest-Shamir-Adleman (RSA) Operation (optional).....	79
6.1.15 Elliptic Curve DSA Operation (optional) .....	80
6.1.16 Elliptic Curve DSA Key Generation (optional) .....	80
6.1.17 Elliptic Curve Diffie-Hellman (ECDH) Key Agreement (optional) .....	81
6.1.18 X25519 (DH with curve25519) (optional) .....	81
6.1.19 Secure Hash Algorithm (SHA) (optional).....	82
6.1.20 Bootloader .....	83
6.1.21 Authentication Proof of Identity.....	85
6.1.22 Protected External Memory .....	85
6.1.23 SMK library operation .....	87
6.1.24 Summary of Security Functional Requirements .....	88
6.2 TOE Assurance Requirements .....	90
6.3 Security Requirements Rationale .....	92
6.3.1 Rationale for the Security Functional Requirements.....	92
6.3.2 Dependencies of Security Functional Requirements .....	98
6.3.3 Rationale for the Assurance Requirements.....	102
6.3.4 Security Requirements are Internally Consistent .....	103

## **7 TOE SUMMARY SPECIFICATION ..... 106**

7.1 List of Security Functional Requirements .....	107
--	-----

## **8 ANNEX ..... 113**

8.1 Glossary .....	113
8.2 Abbreviations .....	115
8.3 References.....	116

# List of Figures

<b>Figure Number</b>	<b>Title</b>	<b>Page Number</b>
Figure 1-1	TOE (STRONGV3P00_In04Ipe) Block Diagram .....	11
Figure 1-2.	Overall Block Diagram of the SoC that includes TOE.....	12
Figure 1-3	Privilege and User Modes .....	16
Figure 1-4	Definition of “TOE Delivery” and responsible Parties.....	21
Figure 3-1	Standard Threats.....	31
Figure 3-2	Threats related to security service .....	31
Figure 3-3	Interactions between the TOE and its outer world .....	32
Figure 4-1	Standard Security Objectives.....	43
Figure 4-2	Security Objectives related to Specific Functionality .....	44

# List of Tables

<b>Table Number</b>	<b>Title</b>	<b>Page Number</b>
Table 1-1	TOE Configuration.....	15
Table 1-2	Method of Delivery.....	16
Table 1-4	Sites of the TOE life cycle .....	19
Table 4-1	Security Objectives versus Assumptions, Threats or Policies .....	54
Table 6-1	Security Functional Requirements defined in Smart Card IC Protection Profile.....	88
Table 6-2	Augmented Security Functional Requirements.....	89
Table 6-3	Security Requirements versus Security Objectives .....	92
Table 6-4	Dependencies of the Security Functional Requirements .....	98
Table 8-1	Abbreviations.....	115

# 1 ST INTRODUCTION

1 This introductory chapter1 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

- 2 The Security Target Lite version is 0.6 and dated 5<sup>th</sup> December 2023.  
The Security Target Lite is strictly compliant to:
- 3 [5] Eurosmart Security IC Platform Protection Profile with Augmentation Packages, Version 1.0, BSI-CC-PP-0084-2014.
- 4 The Protection Profile and the Security Target are built on *Common Criteria version 3.1*.
- Title: Security Target Lite of STRONGV3P00\_In04Ipe of S5E9935 with Specific IC Dedicated Software
  - TOE: Revision 0.0
  - Target of Evaluation: STRONGV3P00\_In04Ipe of S5E9935 with Specific IC Dedicated Software
  - Provided by: Samsung Electronics Co., Ltd.
  - Common Criteria version:
- 5 [1] Common Criteria, Part 1: Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model, Version 3.1, Revision 5, April 2017, CCMB-2017-04-001
- 6 [2] Common Criteria, Part 2: Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components, Version 3.1, Revision 5, April 2017, CCMB-2017-04-002
- 7 [3] Common Criteria, Part 3: Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Components, Version 3.1, Revision 5, April 2017, CCMB-2017-04-003
- 8 [4] Common Methodology for Information Technology Security Evaluation, Evaluation Methodology, Version 3.1, Revision 5, April 2017, CCMB-2017-04-004



## 1.2 TOE Overview and TOE Description

### 1.2.1 Introduction

- 9 The Target of Evaluation (TOE), the STRONGV3P00\_In04Ipe featuring the TORNADO™-H cryptographic coprocessor, is integrated as a Secure Sub System (3S) within an SOC. The TOE is composed of a processing unit, security components, hardware circuits for testing purposes during the manufacturing process and volatile and non-volatile memories (hardware). The TOE also includes IC Designer/Manufacturer proprietary IC Dedicated Software STRONGV3P00\_In04Ipe after being delivered by the IC Manufacturer. Such software is used for providing additional services to facilitate the usage of the hardware, such as a random number generation library for the hardware random number generator. All other software is called STRONGV3P00\_In04Ipe Embedded Software and is not part of the TOE. The SoC S5E9935 is necessary to operate the STRONGV3P00\_In04Ipe but it is not TOE hardware.
- 10 Regarding the AH3 Secure RSA/ECC/SHA library, the user has the possibility to select IC Dedicated Software part of the TOE during the delivery process by choosing their own public key cryptographic library. Hence, the TOE can be delivered with or without the functionality of the AH3 Secure RSA/ECC/SHA library, which results in two TOE configurations. This is considered in this Security Target and corresponding notes (indicated by “optional”) are added where applicable. If the user decides not to use the AH3 Secure RSA/ECC/SHA library, the library is not delivered to the user and the accompanying Rivest-Shamir-Adleman (O.RSA) and Elliptic Curve Cryptography (O.ECDSA, O.ECDH and O.X25519) is not provided by the TOE. Deselecting public key cryptographic libraries means excluding the code that the user decided not to use. Excluding the code of the deselected functionality has no impact on any other security policy of the TOE, it is exactly equivalent to the situation where the user decides just not to use the functionality.

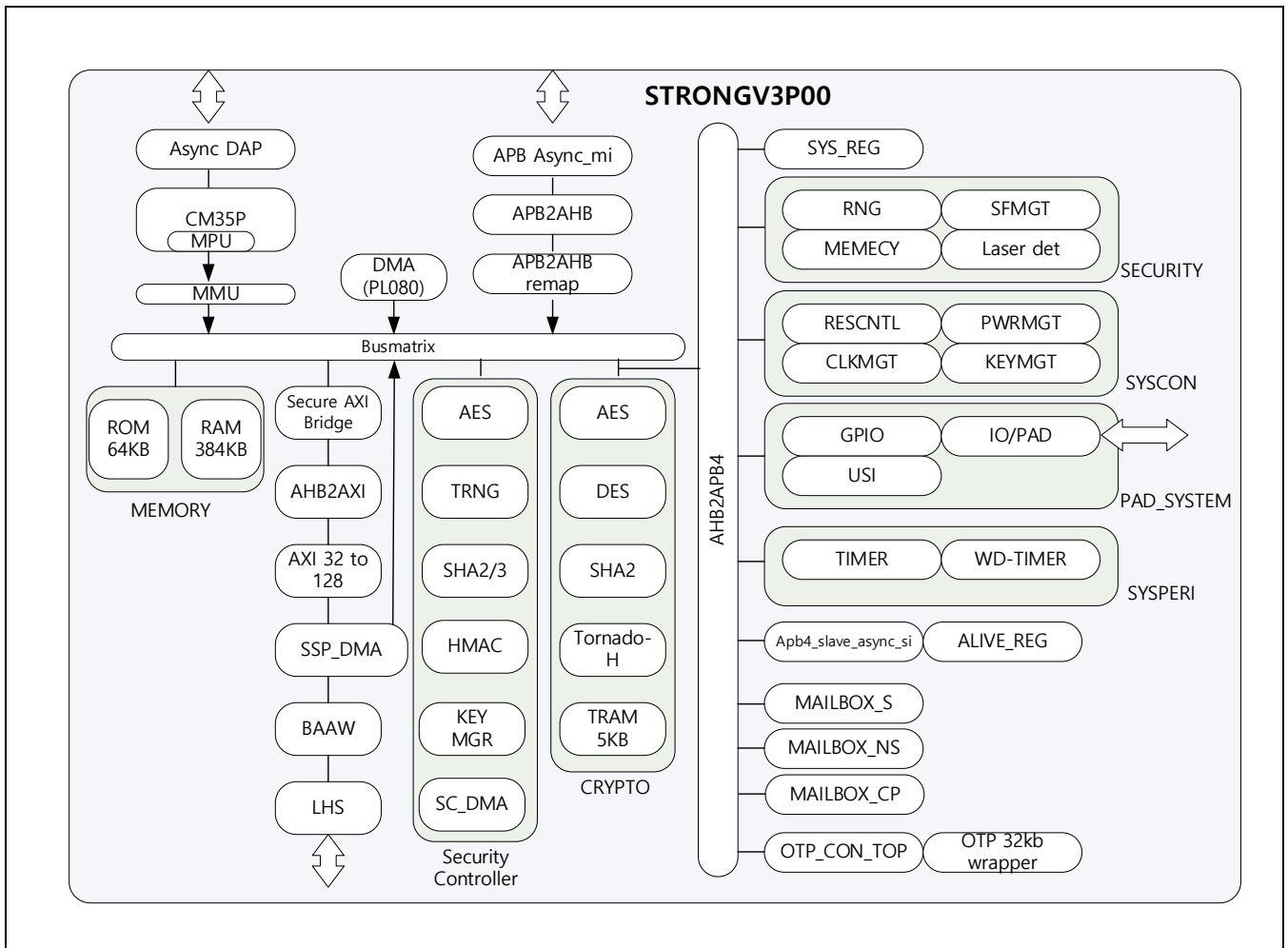
### 1.2.2 TOE Definition

- 11 The TOE is a Secure Sub-System with defined physical boundaries, implemented in a SoC that is designed and packaged specifically for mobile applications.
- 12 The CORTEX-M35P CPU architecture of STRONGV3P00\_In04Ipe follows the Harvard architecture, that is, it has separate program and data memories. Using those separate memory access paths, both instruction and data can be fetched simultaneously without causing a stall.
- 13 The main security features of the **TOE** are:
- Security sensors or detectors
  - Shields
  - Life time detector
  - Dedicated tamper-resistant design based on synthesizable glue logic and secure topology
  - Dedicated hardware mechanisms against side-channel attacks
  - Dedicated hardware mechanisms against Fault Injection attacks
  - Secure TDES and AES Symmetric Cryptography support
  - TORNADO™-H cryptographic coprocessor
  - Key Manager: KDF (block KEYMGR in the Security Controller)
  - ECC/ Parity/ CRC-32 calculators
  - One Hardware Digital True Random Number Generator (DTRNG) that meets PTG.2 class of BSI-AIS-20/31 (German scheme)

- SHA-2/ SHA-3/ HMAC hardware engines in the Security Controller
- Direct Memory Access (SC\_DMA)
- Secure AXI Bridge
- Memory Management Unit (MMU)
- The IC Dedicated Software includes:
  - AH3 Secure RSA/ECC/SHA library for the support of RSA, ECC and SHA cryptographic operations (optional)
  - DTRNG library built around a hardware DTRNG, together with corresponding DTRNG application notes. This library meets PTG.2 class of BSI-AIS-20/31 (German scheme)
  - DRBG library is for deterministic random bit generator as specified in [NIST SP 800-90A][18] using a seed from the DTRNG. This library meets DRG.3 class of BSI-AIS-20/31 (German scheme)
  - Secure Boot Loader is a loader for copying the firmware from an external FLASH storage into the internal SRAM
  - SMK Library supports the encryption keys that are used to encrypt application's encryption keys.

14 The above main security features are part of the evaluation scope.

15 The main hardware blocks of the STRONGV3P00\_In04Ipe Secure Sub-System are described in Figure 1-1 below:

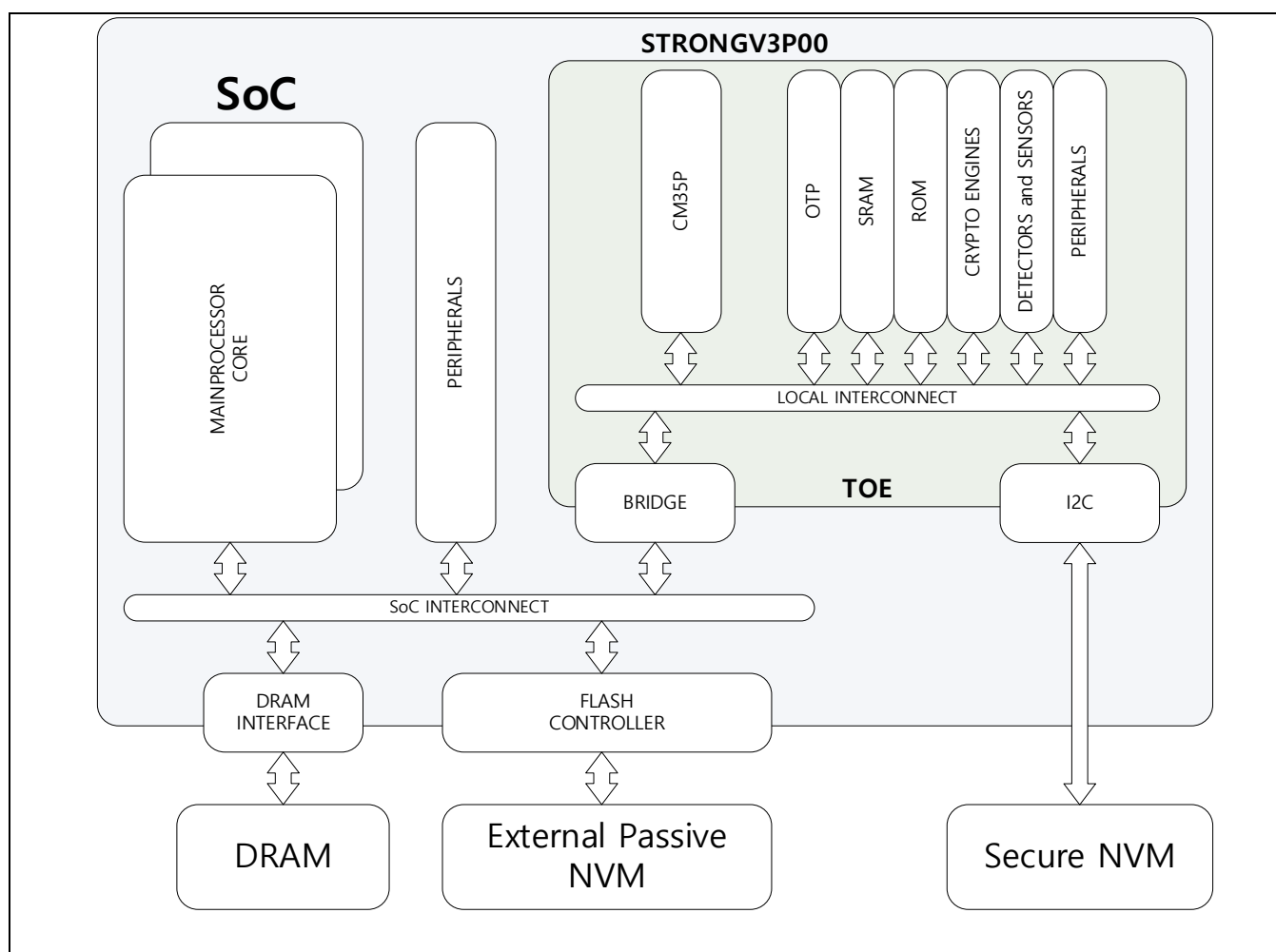


**Figure 1-1 TOE (STRONGV3P00\_In04Ipe) Block Diagram**

**NOTE:** No security functionality is claimed for the following hardware blocks in this TOE or use cases:

- SHA2 in the CRYPTO block
- DMA (PL080)
- SSP\_DMA
- Key manager KEYMGT in SYSCON
- Code execution through the Secure AXI bridge (eXecute In Place, XIP)

**NOTE:** Secure functionality is claimed for the AES in the Security Controller and the AES in the CRYPTO block.



**Figure 1-2. Overall Block Diagram of the SoC that includes TOE**

**NOTE:** The external Secure NVM, DRAM and external Passive NVM are not claimed as a secure functions of this TOE. Additional functionality is required by e.g. an Operating System that implements confidentiality, integrity and rollback protection for the information stored in the external memories.

16 Figure 1-2 shows the overall SoC block diagram. The SoC has multicore CPUs as hosts. In addition, many peripheral IP blocks are connected together through multi-layered SoC interconnect. In the SoC, our TOE is implemented in the form of a logical block. It has a bridge that communicates with the SoC interconnect. The memories such as DRAM or NVM are located outside the SoC.

17 The TOE consists of the following Hardware and Software:

#### TOE Hardware

- 32 Kbit OTP storage/ 384 Kbytes RAM/ 5 Kbytes CryptoRAM (TRAM) / 64 Kbytes ROM
- 32-bit Central Processing Unit (CPU)
- Memory Protection Unit (MPU) with address space up to 4 GB
- Memory Management Unit (MMU)

- Internal Voltage Regulator (IVR)
- Power on Reset circuitry
- Two Internal Clock generators
- Detectors & Security Logic
- Bilateral Pseudo Random Number Generator (BPRNG)
- Digital True Random Number Generator (DTRNG FRO M) that meets PTG.2 class of BSI-AIS-20/31 (German scheme).
- Triple DES cryptographic coprocessor with 112- or 168-bit key size
- Two AES cryptographic coprocessors with 128 bits, 192 bits and 256 bits key size in the Security Controller and CRYPTO block
- Key Manager KDF (block KEYMGR inside the Security Controller)
- TORNADO-H coprocessor, supporting Montgomery multiplication, modular addition/subtraction and a computation for the square of a Montgomery constant up to 4,128-bit operand sizes.
- SHA-2, SHA-3, HMAC hardware engines
- ECC/ Parity/ CRC-32 calculators
- Direct Memory Access (SC\_DMA)
- Secure AXI Bridge
- Timers
- Mailboxes to communicate with the SoC main core

## TOE Software

18 The TOE software comprises the following components:

- The AH3 Secure RSA/ECC/SHA library (optional)  
TORNADO™-H is a hardware coprocessor for high speed modular multiplications, modular additions and modular subtractions. The AH3 Secure RSA/ECC/SHA library is a software library that is built on the TORNADO™-H coprocessor that provides high level interface for RSA, ECC and SHA cryptographic algorithms.

The RSA functions of the library included in the TOE are:

- RSA\_KeyGen\_Secure (RSA public/private key pair generation)
- TND\_RSA\_SigSTD\_Secure (RSA signature generation with the standard method)
- TND\_RSA\_SigCRT\_Secure (RSA signature generation with the CRT method)
- TND\_RSA\_Verify (RSA signature verification)
- RSA\_R2modM\_precompute\_sec ( $R^2$  value precomputation for the standard RSA)
- RSA\_R2modPandQ\_precompute\_sec ( $R^2$  value precomputation for the CRT RSA)

The library supports RSA operations with key sizes from 32-bits to 4,096-bits, in 2-bit steps. However, only the key size range from 1,900-bit up to 4,096-bit is within the scope of this evaluation.

The functions TND\_RSA\_SigSTD\_Secure and TND\_RSA\_SigCRT\_Secure implement countermeasures against SPA, DPA and DFA attacks. The RSA\_KeyGen\_Secure function implements countermeasures against SPA and DFA attacks. Finally, the RSA\_R2modM\_precompute\_sec and

RSA\_R2modPandQ\_precompute\_sec functions implement countermeasures against Fault Injection attacks.

The AH3 Secure RSA/ECC/SHA library provides a set of functions to implement ECC cryptographic algorithms. In particular, it provides functions to implement the ECDSA signing/verifying and the ECDH key exchange protocol. The library implements ECC for general curves over prime fields of sizes from 224-bit to 512-bit. Only curves whose security has been proven are in scope of this evaluation (see Note 1 below). The ECC functions of the library included in the TOE are:

- ECDSA\_keygen (Ephemeral or static key pair generation for ECDSA signing/verifying)
- ECDSA\_sign\_digest (ECDSA signature generation for a message digest)
- ECDSA\_verify\_digest (ECDSA signature verification for a message digest)
- ECDH\_generate (ECDH secret key derivation)
- X25519 (DH with curve 25519)
- X25519\_with\_decoded\_number (X25519 with decoded scalar)

The functions ECDSA\_keygen, ECDSA\_sign\_digest and ECDH\_generate implement countermeasures against SPA, DPA and DFA for protection of the private key. The function ECDSA\_verify\_digest implements countermeasures against DFA. The base point is assumed to be public.

Note1) The AH3 Secure RSA/ECC/SHA library supports any valid elliptic curves over prime fields of sizes from 224-bit to 512-bit. However, only the proven standard curves listed below are in the scope of this evaluation.

- 1) *[NIST curves]*: Curves P-224, P-256, P-384, P-521
- 2) *[Brainpool curves]*: brainpoolP224r1, brainpoolP224t1, brainpoolP256r1, brainpoolP256t1, brainpoolP320r1, brainpoolP320t1, brainpoolP384r1, brainpoolP384t1, brainpoolP512r1, brainpoolP512t1
- 3) *[SEC-recommended curves]*: secp224k1, secp224r1, secp256k1, secp256r1, secp384r1, secp521r1
- 4) *[RFC7748]*: Curve25519

The AH3 Secure RSA/ECC/SHA library provides functions for calculating hash (digest) values using the SHA1, SHA224, SHA256, SHA384 and SHA512 algorithms as specified in [FIPS PUB 180-3]. This library implements the following functions for the algorithms SHA224, SHA256, SHA384 and SHA512:

- SHA224\_init, SHA224\_update, SHA224\_final
- SHA256\_init, SHA256\_update, SHA256\_final
- SHA384\_init, SHA384\_update, SHA384\_final
- SHA512\_init, SHA512\_update, SHA512\_final

These functions which is in AH3 Secure RSA/ECC/SHA library do claim protection against Fault attacks, but do not claim protection against side channel analysis attacks.(i.e. these functions shall not be used to hash confidential information, such as keys etc.) These functions shall be only used for message digest of ECDSA and RSA digital signature. These functions shall not be used for other purposes.

- A Digital True Random Number Generator library (DTRNG FROM library) that fulfills the requirements of Class PTG.2 of BSI-AIS-20/31 (German Scheme).

- A Deterministic Random Bit Generator library (DRBG library) that fulfills the requirements of Class DRG.3 of BSI-AIS-20/31 (German Scheme).
- The Secure Boot Loader is a loader for copying, authenticating and decrypting firmware from an external FLASH storage into the internal SRAM.
- The SMK Library supports the encryption key for the application encryption key.  
Note: The SMK library will only be made available for the OS developer, because it provides a support function that will only be of use for the OS developer.
- The TOE configuration is summarized in Table 1-1 below:

Table 1-1 TOE Configuration

Item type	Item	Version	Format	Form of delivery
Hardware	STRONGV3P00_In04lpe Secure Sub-System on the S5E9935 SoC	0.0	-	Hardware Secure Sub System as part of a SoC in a Package-on-Package (PoP)
Hardware	SoC Package	1462-FOWLP-14.0x15.3	-	PoP with DRAM
Hardware	SoC S5E9935, embedding the TOE	0.0	-	SOC in a PoP
Software	Secure Boot loader	1.2	-	Stored in ROM of the STRONGV3P00_In04lpe
Software	AH3 Secure RSA/ECC/SHA Library (optional)	1.06	-	Software Library. This library is delivered as object file and is optionally integrated into user code.
Software	DTRNG library	1.1	-	Software Library. This library is delivered as object file and is optionally integrated into user code.
Software	DRBG library	1.04	-	Software Library. This library is delivered as object file and is optionally integrated into user code.
Software	SMK library	0.1	-	Software Library. This library is delivered as object file and is optionally integrated into user code.
Document	STRONGV3P00_In04lpe HW DTRNG FRO M and DTRNG FRO M Library Application Note	1.3	PDF	Softcopy
Document	STRONGV3P00_In04lpe DRBG library Application Note	1.2	PDF	Softcopy
Document	AH3 Secure RSA /ECC/SHA Library API Manual	1.06	PDF	Softcopy
Document	STRONGV3P00_In04lpe SMK Library Application Note	0.1	PDF	Softcopy

Item type	Item	Version	Format	Form of delivery
Document	STRONGV3P00 of S5E9935, 32-bit RISC Microcontroller for Secure Element Platform	0.7	PDF	Softcopy
Document	Security Application Note for STRONG_V3P00_In04lpe	1.5	PDF	Softcopy
Document	S5E9935 Chip Delivery Specification	0.3	PDF	Softcopy
Document	STRONGV3P00_In04lpe Secure Bootloader Manual for S5E9935	0.2	PDF	Softcopy
Document	CORTEX-M35P Reference manual	0.0	PDF	Softcopy

Table 1-2 Method of Delivery

Item	Method of delivery
Hardware	Secure carrier
Software	Libraries are encrypted by PGP encryption and then delivered by email.
Documents	Documents are encrypted by PGP encryption and then delivered by e-mail.

## 19 PRIVILEGE mode and USER mode:

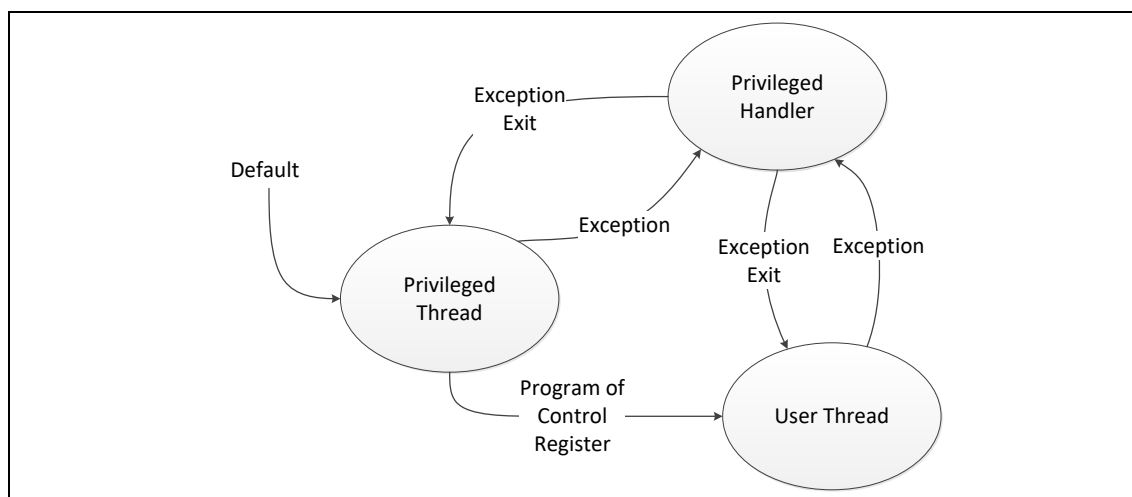


Figure 1-3 Privilege and User Modes

- 20 Code can execute as privileged or unprivileged (user).
- 21 Software running at privileged access level can switch the program into user access level using the control register. When an exception takes place, the processor will always switch back to privileged mode and return to the previous mode when exiting the exception handler.



### 1.2.3 TOE Features

#### 22 CPU

- Cortex-M35P 32-bit core (MPU extension up to 4GB)

#### 23 Memory

- 64 KB MASK ROM (containing the Samsung built-in Secure Boot Loader)
- 384 KB SRAM (general purpose; firmware and data)
- 5 KB CryptoRAM (dedicated for TORNADO-H operation, therefore the user cannot use this area for general purpose data storage)
- 32 K-bit secure One Time Programmable memory (OTP)

#### 24 DES

- Built-in hardware DES accelerator
  - Circuits for resistance against side channel and Fault Injection attacks
  - ECB mode
- \* Only Triple DES allowed.

#### 25 AES in “CRYPTO block” and “Security Controller”

- Built-in hardware AES accelerators
- Circuits for resistance against side channel and Fault Injection attacks
- ECB mode
- CBC mode
- CTR mode
- GCM mode

#### 26 TORNADO-H

- TORNADO-H coprocessor, supporting Montgomery multiplication, modular addition/subtraction and a computation for the square of a Montgomery constant up to 4,128-bit operand sizes

#### 27 Abnormal Condition Detectors

- Environmental & Life Time Detector

#### 28 Interrupts

- Nested Vector Interrupt Controller

## 29 Reset and Power Down Mode

- Power-on reset and reset sequencer
- Power can be turned off by an external power management unit (Power Down Mode)

## 30 Random Number Generator

- A Digital True Random Number Generator (DTRNG): PTG.2 class, compliant to BSI-AIS-20/31 (German Scheme)
- A Bilateral Pseudo Random Number Generator (BPRNG): no compliance to any specific metric, but BPRNG is used by the chip internally and for security software countermeasures. It is to be seeded by the DTRNG.

## 31 Memory Protection Unit

- Memory Protection Unit (MPU) with an access space of up to 4 GB

## 32 Memory Management Unit

- MMU performs address translation to map physical addresses (PA) to virtual addresses (VA) without setting individual protection attributes for each partition.

## 33 Memory Encryption and Bus Scrambling

- Static bus scrambling
- Dynamic Data encryption/ decryption of buses
- Automatic ROM encryption/ decryption
- Automatic RAM encryption/ decryption

## 34 Timers

- 32-Bit Timer programmable interval timers
- 20-bit Watchdog Timer

## 35 CRC

- 32bit - CRC32

## 36 Clock Sources

- Internal clock: up to 300 MHz
- RTC

- 37 HASH engine
  - SHA256/384/512 based on HASH standard-NIST FIPS 180-4
  - SHA3 / SHAKE based on HASH standard-NIST FIPS PUB 202
  - SHA1-based / SHA2-based / SHA3-based HMAC
- 38 Operating Voltage Range
  - 1.2V+-5%
- 39 Operating Temperature
  - - 25°C to 85°C
- 40 Power inputs physically isolated from the SoC power supplies
- 41 Mailboxes to communicate with external components (application processor, external memories)
- 42 Secure DMA, inside the Security Controller block (SC\_DMA)
- 43 Secure AXI Bridge
- 44 Package on Package
  - DRAM packaged on top of the SoC package

#### 1.2.4 TOE Life cycle

- 45 The complex development and manufacturing processes of a Composite Product can be separated into seven distinct phases. The phases 2, 3 and 4 of the Composite Product life cycle cover the IC development and production of this TOE:

**Table 1-3 Sites of the TOE life cycle**

Site / Building	Phase
Hwasung Plant/ DSR Building	Phase 2
Giheung Plant/ SR3 Building	Phase 2
Hwasung Plant/ Line S3	Phase 3

Giheung Plant/ SR1 Building	Phase 3
Hwasung Plant/ MR2 (NRD) Building	Phase 3
Giheung Plant/ Line 5	Phase 3
Giheung Plant/ Line 3	Phase 3
Giheung Plant/ Line 2	Phase 3
TESNA Plant	Phase 3
Onyang Plant/ Warehouse	Phase 4
Onyang Plant/ Line 2	Phase 3+4

- 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

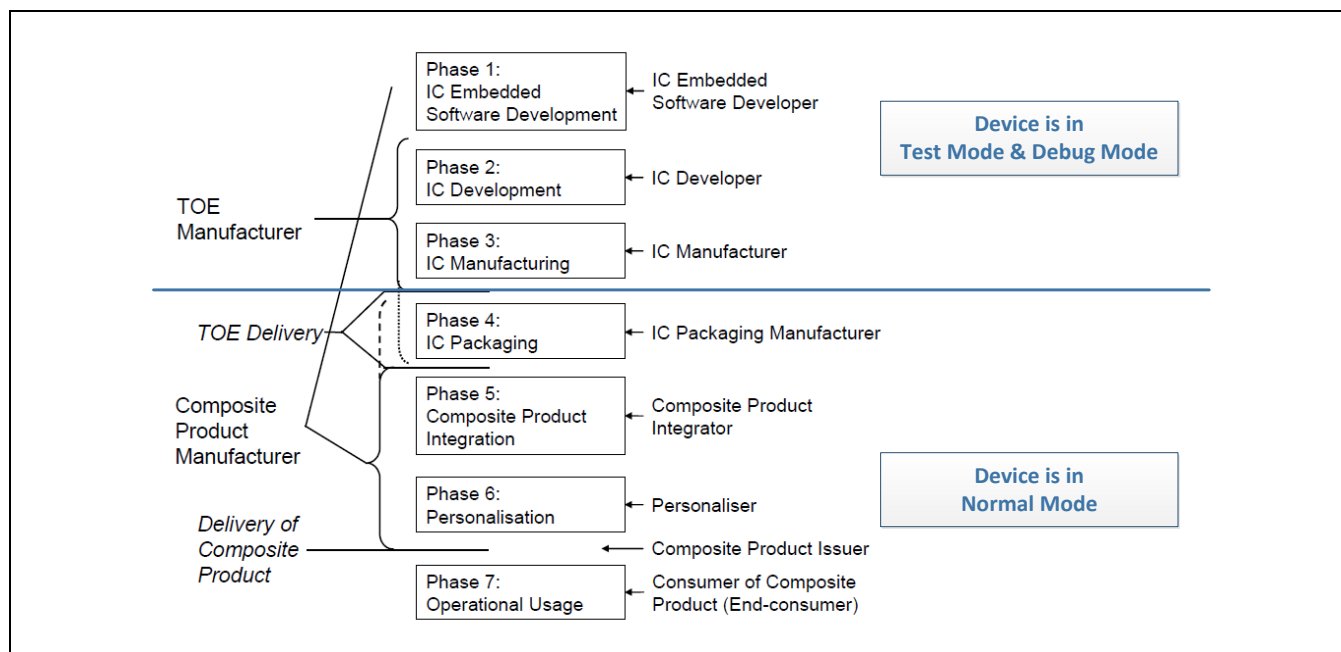
\* The fusing from TEST mode to NORMAL mode is the Phase3 which is in IC testing (EDS step).

46 The Composite Product life cycle phase 4 is included in the evaluation of the IC:

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

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

- Security IC Embedded Software Development (Phase 1),
- the Composite Product finishing process, preparation and shipping to the personalisation line for the Composite Product (Composite Product Integration Phase 5),
- the Composite Product Personalisation and testing stage where the User Data 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 1-4 Definition of "TOE Delivery" and responsible Parties**

- 48 The Security IC Embedded Software is developed outside the TOE development in Phase 1. The TOE is developed in Phase 2 and produced in Phase 3. The TOE is packaged in Phase 4.

### 1.3 Interfaces of the TOE

49 TOE has the following interfaces:

- The physical interface of the TOE with the external environment is the entire surface of the STRONGV3P00\_In04Ipe
- The electrical interfaces of the TOE with the external environment are AVDD12\_LDO\_STR, AVDD18\_LDO\_STR, VDD\_ALIVE, AVSS\_LDO\_STRONG.
- The data interfaces of the TOE consists of Mailboxes, Secure DMA (SC\_DMA), I2C to Secure Flash and the Secure AXI Bridge (BAAW, Long Hop (LHS))
- The software interfaces of the TOE with the hardware consist of Special Function Registers (SFR) and CPU instructions
- The DTRNG interface of the TOE is defined by the DTRNG libraries interface
- The DRBG interface of the TOE is defined by the DRBG library interface
- The PKA interface of the TOE is defined by the AH3 Secure RSA/ECC/SHA library interface (optional).
- The SMK interface of the TOE is defined by the SMK library interface (optional).
- The Secure Boot Loader interface

## 1.4 TOE Intended Usage

50 The TOE is intended to be used for applications that require security functions like:

- Cryptographic operations such as AES encryption and decryption, TDES encryption and decryption, big-number calculations and random number generation
- Cryptographic key import and secure storage inside the TOE

# 2 CONFORMANCE CLAIMS

51 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

- 52 This Security target claims to be conformant to the Common Criteria version 3.1 R5.
- 53 Furthermore it claims to be CC Part 2 extended and CC Part 3 conformant. The extended Security Functional Requirements are defined in chapter 5.
- 54 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 5, April 2017, CCMB-2017-04-001
  - [2] Common Criteria, Part 2: Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components, Version 3.1, Revision 5, April 2017, CCMB-2017-04-002
  - [3] Common Criteria, Part 3: Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Components, Version 3.1, Revision 5, April 2017, CCMB-2017-04-003
  - [4] Common Methodology for Information Technology Security Evaluation, Evaluation Methodology, Version 3.1, Revision 5, April 2017, CCMB-2017-04-004

## 2.2 PP Claim

- 55 This Security Target is strictly compliant to the Security IC Platform Protection Profile [5]. The Security IC Platform Protection Profile is registered and certified by the Bundesamt für Sicherheit in der Informationstechnik (BSI) under the reference BSI-CC-PP-0084, Version 1.0, dated 01.2014
- 56 This ST does not claim conformance to any other PP.

## 2.3 Package Claim

- 57 The assurance level for this Security Target is EAL5 augmented with AVA\_VAN.5 and ALC\_DVS.2.
- 58 This Security Target is strictly compliant to the Security IC Platform Protection Profile [5] with additional packages:
- Package “Authentication of the Security IC”
  - Package 2: Loader dedicated for usage by authorized users only

## 2.4 Conformance Claim Rationale

- 59 This security target claims strict conformance only to one PP, the Security IC Platform Protection Profile [5].
- 60 The Evaluation Assurance Level (EAL) of the PP [5] is EAL 5, augmented with the assurance components ALC\_DVS.2 and AVA\_VAN.5. The Assurance Requirements of the TOE obtain the Evaluation Assurance Level 5, augmented with the assurance components ALC\_DVS.2 and AVA\_VAN.5.
- 61 The Target of Evaluation (TOE) is a complete solution, implementing a secure integrated circuit (secure IC) as defined in the PP [5], section 1.2.2, so the TOE is consistent with the TOE type in the PP [5].
-

- 62 The security problem definition of this security target is consistent with the statement of the security problem definition in the PP [5], as the security target claims strict conformance to the PP [5]. Additional threats, organizational security policies and assumptions are introduced in chapter 3 of this ST. A rationale is given in chapter 4.4.
- 63 The security objectives of this security target are consistent with the statement of the security objectives in the PP [5], as the security target claims strict conformance to the PP [5]. Additional security objectives are added in chapter 4.1 of this ST. A rationale is given in chapter 4.4.
- 64 The security requirements of this security target are consistent with the statement of the security requirements in the PP [5], as the security target claims strict conformance to the PP [5]. Additional security requirements are added in chapter 6.1 of this ST. A rationale is given in chapter 6.3. All assignments and selections of the security functional requirements are done in the PP [5] and in section 6 of this security target.

# 3 SECURITY PROBLEM DEFINITION

65 This chapter 3 contains the following sections:

3.1 Description of Assets

3.2 Threats

3.3 Organizational Security Policies

3.34 Assumptions

### 3.1 Description of Assets

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

- the User Data of the Composite TOE,
- the Security IC Embedded Software stored and in operation,
- the Security Services provided by the TOE for the Security IC Embedded Software.

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

- SC1 integrity of user data of the Composite TOE,
- SC2 confidentiality of user data of the Composite TOE being stored in the TOE's protected memory areas,
- SC3 correct operation of the security services provided by the TOE for the Security IC Embedded Software.

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

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

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

70 If User Data is stored in external FLASH memory, the security IC shall protect it in confidentiality before exporting it outside the TOE Hardware and storing it in external FLASH memory. The security IC shall implement security mechanisms to protect User Data of the Composite TOE in integrity, confidentiality, authenticity and actuality if stored in external FLASH memory. It is considered as a security service provided by the TOE for the Security IC Embedded Software.

71 The Protection Profile [5] requires the TOE to provide at least one security service: the generation of random numbers by means of a physical Random Number Generator. The Security Target may require additional security services as described in the annex 7 of the protection profile or define TOE specific security services. It is essential that the TOE ensures the correct operation of all security services provided by the TOE for the Security IC Embedded Software.

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

- SC4 deficiency of random numbers.

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

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

- 74 Such information and the ability to perform manipulations assist in threatening the above assets.
- 75 Note that there are many ways to manipulate or disclose the user data of the Composite TOE: (i) An attacker may manipulate the Security IC Embedded Software or the TOE. (ii) An attacker may cause malfunctions of the TOE or abuse Test Features provided by the TOE. Such attacks usually require design information of the TOE to be obtained. They pertain to all information about (i) the circuitry of the IC (hardware including the physical memories), (ii) the IC Dedicated Software with the parts IC Dedicated Test Software (if any) and IC Dedicated Support Software (if any), and (iii) the configuration data for the TSF. The knowledge of this information may enable or support attacks on the assets. Therefore the TOE Manufacturer must ensure that the development and production of the TOE (refer to Section 1.2.4) is secure so that no restricted, sensitive, critical or very critical information is unintentionally made available for attacks in the operational phase of the TOE (cf. [8] for details on assessment of knowledge of the TOE in the vulnerability analysis).
- 76 The TOE Manufacturer must apply protection to support the security of the TOE. This not only pertains to the TOE but also to all information and material exchanged with the developer of the Security IC Embedded Software. This covers the Security IC Embedded Software itself if provided by the developer of the Security IC Embedded Software or any authentication data required to enable the download of software. This includes the delivery (exchange) procedures for Phase 1 and the Phases after TOE Delivery as far as they can be controlled by the TOE Manufacturer. These aspects enforce the usage of the supporting documents and the refinements of SAR defined in the protection profile.
- 77 The information and material produced and/or processed by the TOE Manufacturer in the TOE development and production environment (Phases 2 up to TOE Delivery) can be grouped as follows:
- logical design data,
  - physical design data,
  - IC Dedicated Software, Initialisation Data and Pre-personalisation Data,
  - Security IC Embedded Software, provided by the Security IC Embedded Software developer and implemented by the IC manufacturer,
  - specific development aids,
  - test and characterisation related data,
  - material for software development support, and
  - photomasks and products in any form
- 78 as long as they are generated, stored, or processed by the TOE Manufacturer.

## 3.2 Threats

- 79 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 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 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.
- 80 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. This should be considered for the threat T.Masquerade\_TOE.
- 81 The cloning of that functional behaviour requires to (i) develop a functional equivalent of the Security IC Embedded Software, (ii) disclose, interpret and employ the user data of the Composite TOE stored in the TOE, and (iii) develop and build a functional equivalent of the Security IC using the input from the previous steps.
- 82 The Security IC is a platform for the Security IC Embedded Software which ensures that especially the critical user data of the Composite TOE are stored and processed in a secure way (refer to below). The Security IC Embedded Software must also ensure that critical user data of the Composite TOE are treated as required in the application context. In addition, the personalisation process supported by the Security IC Embedded Software (and perhaps by the Security IC in addition) must be secure. This last step is beyond the scope of this security target. As a result the threat “cloning of the functional behaviour of the Security IC on its physical and command interface” is averted by the combination of mechanisms which split into those being evaluated according to this security target (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.
- 83 The high-level security concerns are refined below by defining threats as required by the Common Criteria (refer to Figure 3-1 Standard Threats). Note that manipulation of the TOE is only a means to threaten user data and is not a success for the attacker in itself.

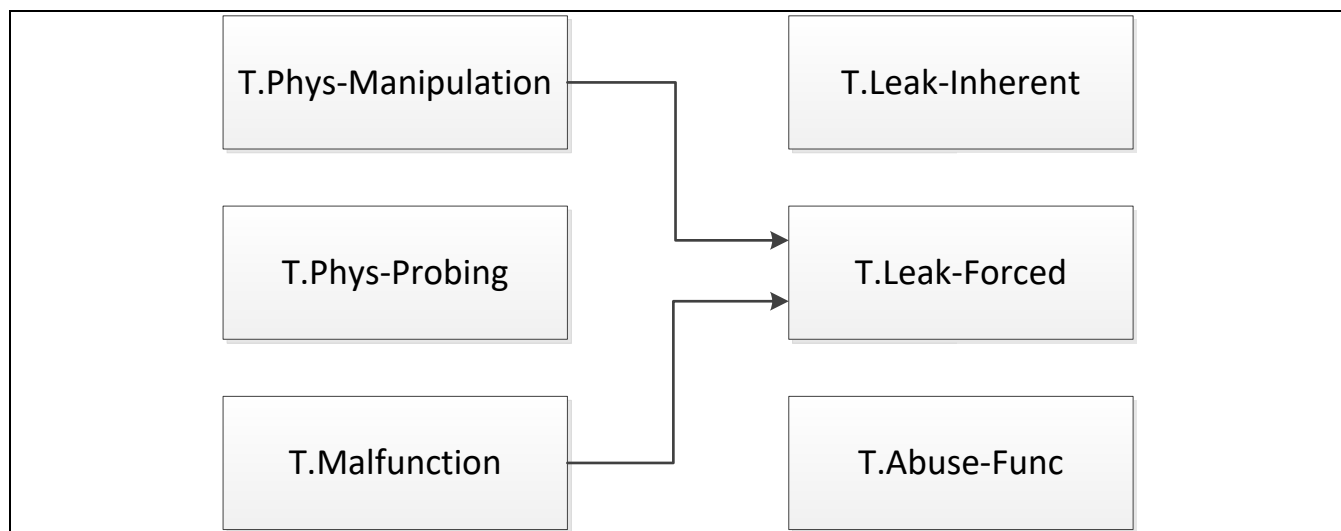


Figure 3-1 Standard Threats

- 84 The high-level security concern related to security service is refined below by defining threats as required by the Common Criteria (refer to Figure 3-2 Threats related to security service).

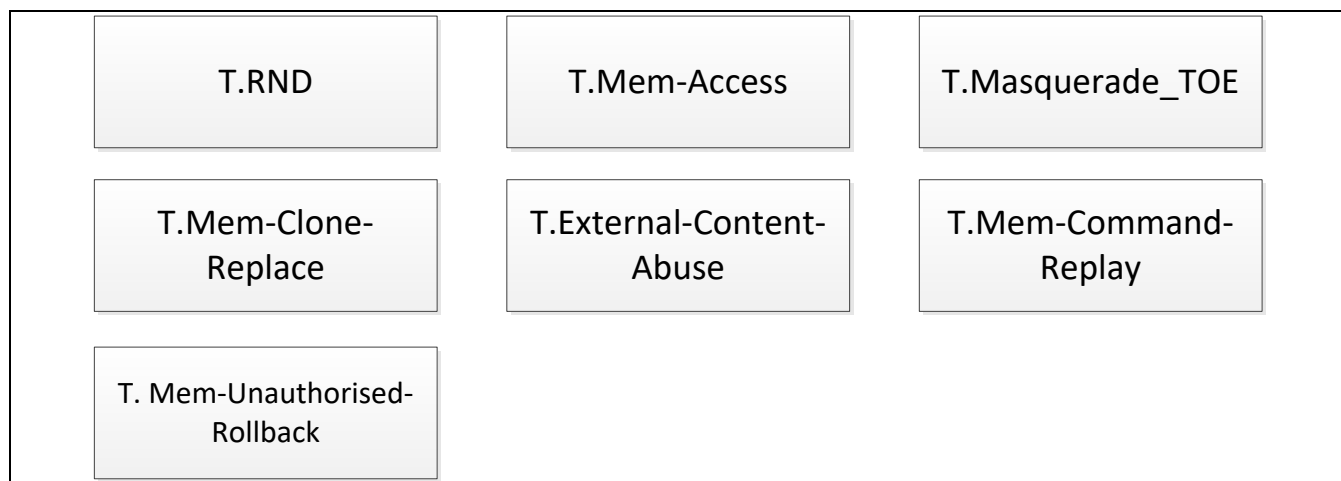


Figure 3-2 Threats related to security service

- 85 The Security IC Embedded Software must contribute to averting the threats: At least it must not undermine the security provided by the TOE.
- 86 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.
- 87 The TOE's countermeasures are designed to avert the threats described below. Nevertheless, they may be effective in earlier phases (Phases 4 to 6).

- 88 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 3-3. Due to the intended usage of the TOE all interactions are considered as possible.

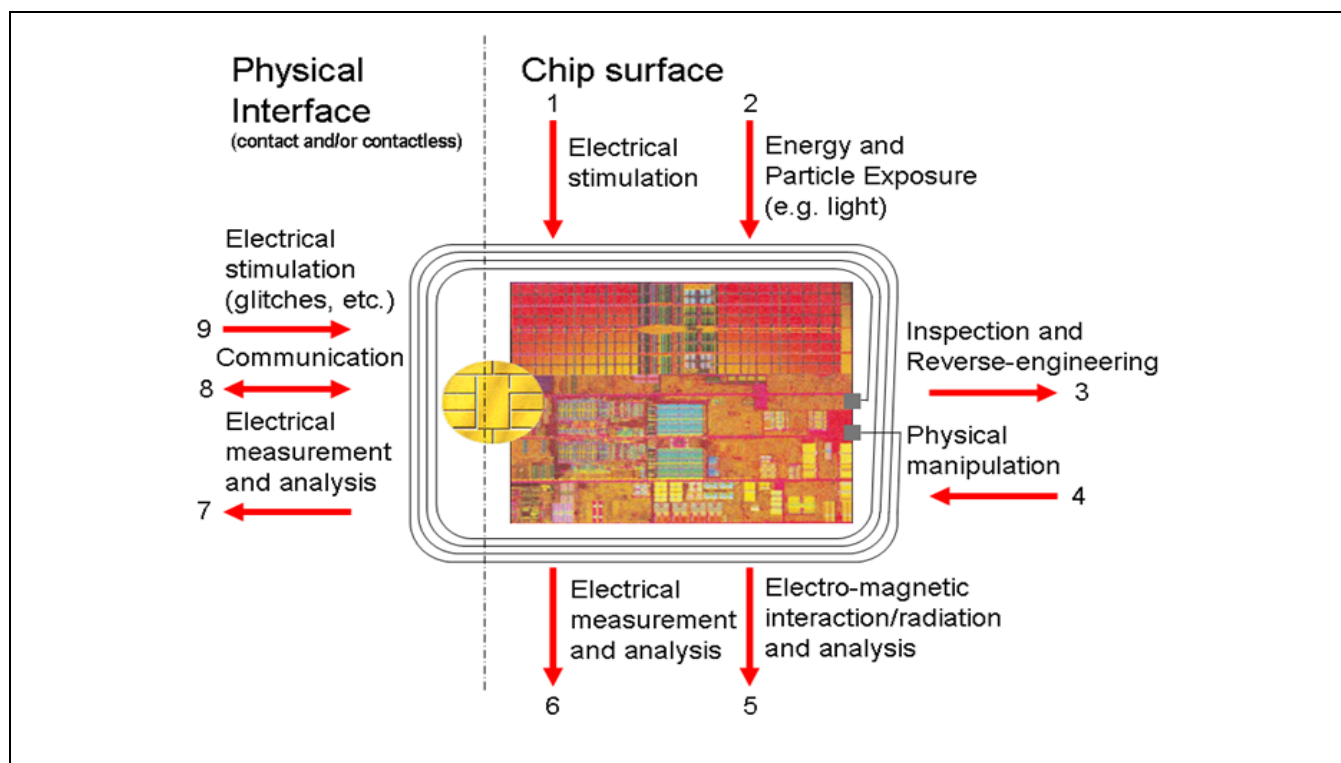


Figure 3-3 Interactions between the TOE and its outer world

- 89 An interaction with the TOE can be done through the physical interfaces (Number 7 – 9 in Figure 3-3) which are realised using contacts and/or a contactless interface. Influences or interactions with the TOE also occur through the chip surface (Number 1 – 6 in Figure 3-3). 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.

### 3.2.1 Standard Threats

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

T.Leak-Inherent      Inherent Information Leakage



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

91 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 3-3) or measurement of emanations (Number 5 in Figure 3-3) and can then be related to the specific operation being performed.

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

T.Phys-Probing      Physical Probing

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

93 Physical probing requires direct interaction with the Security IC internals (Numbers 5 and 6 in Figure 3-3). 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 3-3). Determination of software design including treatment of user data of the Composite TOE may also be a pre-requisite.

94 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.

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

T.Malfunction      Malfunction due to Environmental Stress

An attacker may cause a malfunction of TSF or of the Security IC Embedded Software by applying environmental stress in order to (i) modify security services of the TOE or (ii) modify functions of the Security IC Embedded Software (iii) deactivate or affect security mechanisms of the TOE to enable attacks disclosing or manipulating the user data of the Composite TOE or the Security IC Embedded Software. This may be achieved by operating the Security IC outside the normal operating conditions (Numbers 1, 2 and 9 in Figure 3-3).

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

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

T.Phys-Manipulation      Physical Manipulation

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

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

99 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 3-3).

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

T.Leak-Forced                      Forced Information Leakage

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

101 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 3-3) which normally do not contain significant information about secrets.

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

T.Abuse-Func                      Abuse of Functionality

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

### 3.2.2 Threats related to security services

103 The TOE shall avert the threat "Deficiency of Random Numbers (T.RND)" as specified below.

T.RND                                      Deficiency of Random Numbers

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

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

### 3.2.3 Threats related to additional TOE Specific Functionality

104 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 Security IC 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 Security 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.

### 3.2.4 Threats related to Authentication of the Security IC

105 The TOE shall avert the threat “Masquerade the TOE (T. Masquerade\_TOE)” as specified below.

T.Masquerade\_TOE              Masquerade the TOE

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

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

### 3.2.5 Threats related to External Memory

106 The TOE shall avert the threat “Cloning the TOE with a Copy of the external memory (T.Mem-Clone-

Replace)” as specified below.

T.Mem-Clone-Replace

Cloning or replacement of external memory

An attacker may attempt to clone the full content of the external memory or a specific memory area storing User Data of the 3S and write it to the external memory used by a different unit; alternatively, an attacker may physically replace the external memory used by a 3S with a different memory that may come from a different unit.

This threat refers to the case where partial or full content of the external memory is cloned to a different device. It can also cover the replacement of the physical external memory used by the 3S with the memory of a different unit. The second case might not be viable on some architectures or memory when the physical design or assembly procedures impede it.

The effect of this threat is in replacing the data and/or image of a TOE with a different one and to obtain a valid but unauthorised instance of the TOE.

This threat involves using two different TOE units or instances. One TOE unit is used as a source for the external memory content. This content is used to replace the genuine content of the external memory of the second TOE unit.

Another possible scenario for this threat can be contemplated for passive external non-volatile memory: the external non-volatile memory is replaced with an empty or virgin non-volatile memory, removing the user and TSF data used by the TOE, and possibly forcing the TSF to generate new user and TSF data, potentially affecting the TSF behaviour.

107 The TOE shall avert the threat “Abuse of external memory content (T.External-Content-Abuse)” as specified below.

T.External-Content-Abuse

Abuse of external memory content

An attacker may attempt to access for disclosing or modifying the content of the external memory used by the 3S. Thereby an attacker may compromise confidentiality and/or integrity of TSF data and/or user data that shall be protected by the TOE.

An attacker may obtain unauthorised access to the external memory and attempt to read, disclose, modify or replace the content of the external memory. This threat addresses also the authenticity of the data stored in the external memory.

Note that the access to the external memory or the transfer of data between the TOE and the external memory may also support an attack.

108 The TOE shall avert the threat “Replay of commands between the 3S and the external memory (T.Mem-Command-Replay)” as specified below.

T.Mem-Command-Replay

Replay of commands between the 3S and the external memory

An attacker may attempt to replay the write and erase commands or responses to the read commands between the 3S and the passive external

memory, to affect the freshness of the content read from or written to the external memory.

The read, write and erase commands issued by the 3S to exercise the storage functionality of the external memory, and their payloads, can be intercepted by an attacker (e.g. eavesdrop the commands on the link between the 3S and the external memory). Such attacker may use copies of these commands to try to misuse the TOE or compromise data. The command replay attack can take the following forms:

- The attacker reacts on a read command and replies a previously recorded answer e.g. to a previous read request. Thereby the 3S gets an old version of such content.
- The attacker issues a previous write command, trying to overwrite the external memory with the previous content, leading to the 3S obtaining old versions of such content in later read operations.
- The attacker issues a previous erase command, trying to overwrite status information or other data that may lead to misuse for the TOE.

109 The TOE shall avert the threat “Unauthorised rollback of content in the external memory (T.Mem-Unauthorised-Rollback)” as specified below.

T.Mem-Unauthorised-Rollback

Unauthorised rollback of content in the external memory

An attacker may attempt to read the content of the external memory, record it, and later write it back to the external memory after the original content were updated by the TOE.

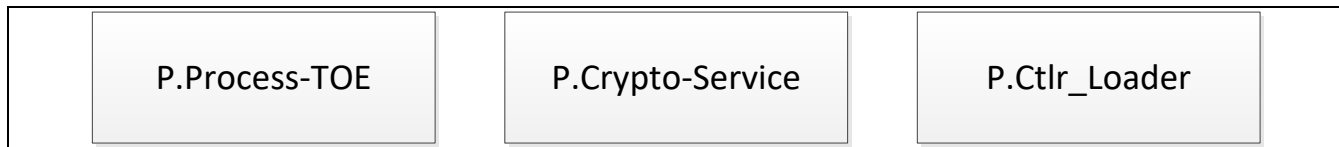
This threat takes advantage of the fact that the external memory is not integrated into the 3S. Hence, physical protections for preventing the replacement of content may not cover the external memory. This situation enables an attacker to read and write the content of the external memory. Even if the confidentiality and integrity of the external memory content is protected, the replacement with an old copy may be valid as well, since it is retrieved from the external memory.

If the TOE image is stored in an external non-volatile memory, this threat may lead to an unauthorised rollback of the TOE image to an older version. Even when the TOE stores data and not code in the external memory, this data rollback might affect the behaviour of the TSF.

The replacement of content stored in the external memory with previous versions of it may refer to the full content of the external memory or partial content of it, depending on the organization and protection of the data stored in the external memory.

### 3.3 Organizational Security Policies

110 The following Figure 3-5 Organizational Security Policies shows the policies applied in this Security Target.



**Figure 3-5 Organizational Security Policies**

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

P.Process-TOE                      Identification during TOE Development and Production

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

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

113 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.

114 The TOE provides specific cryptographic services which can be used by the Security IC Embedded Software. In the following specific cryptographic services are listed which is not derived from threats identified for the TOE’s environment because it can only be decided in the context of the Security IC applications, against which threats the Security IC Embedded Software will use the specific cryptographic service.

The IC Developer / Manufacturer must apply the policy “Cryptographic Service (P.Crypto-Service)” as specified below.

P.Crypto-Service                      Cryptographic Services provided by the TOE

The TOE shall provide the following cryptographic services to the IC Embedded Software:

- (Triple) Data Encryption Standard (DES)
- Advanced Encryption Standard (AES)
- Rivest-Shamir-Adleman (RSA) public key asymmetric cryptography (optional)
- Elliptic Curve Cryptography (ECC) (optional)
- Secure Hash Algorithm (SHA) (optional)
- Secure Hash Algorithm (SHA\_HW)

Note: The TOE can be delivered without the AH3 Secure RSA/ECC/SHA library. In this case the TOE does not provide the Additional Specific Security Functionality Rivest-Shamir-Adleman Cryptography and Elliptic Curve Cryptography (ECC) and the software Secure Hash Algorithm (SHA).

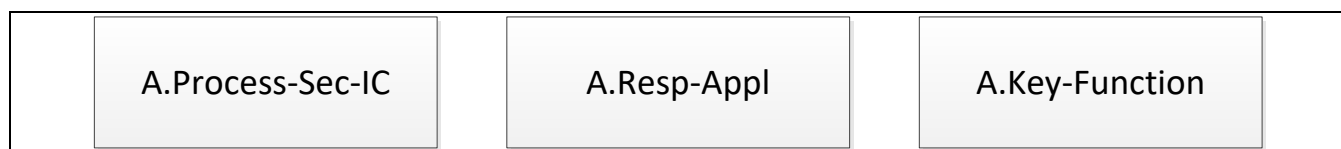
The organizational security policy “Controlled usage to Loader Functionality (P.Ctlr\_Loader)” applies to Loader dedicated for usage by authorized users only.

P.Ctlr\_Loader                      Controlled usage to Loader Functionality

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

### 3.4 Assumptions

115 The following shows the assumptions applied in this Security Target.



**Figure 3-4 Assumptions**

- 116 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.
- 117 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.
- 118 Appropriate “Protection during Packaging, Finishing and Personalisation (A.Process-Sec-IC)” must be ensured after TOE Delivery up to the end of Phase 6, as well as during the delivery to Phase 7 as specified below.

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

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

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

- 119 The information and material produced and/or processed by the Security IC Embedded Software Developer in Phase 1 and by the Composite Product Manufacturer can be grouped as follows:
- the Security IC Embedded Software including specifications, implementation and related documentation,
  - Pre-personalisation Data and Personalisation Data including specifications of formats and memory areas, test related data,
  - the user data of the Composite TOE and related documentation, and
  - material for software development support
- 120 as long as they are not under the control of the TOE Manufacturer. Details must be defined in the Protection Profile or Security Target for the evaluation of the Security IC Embedded Software and/or Security IC.
- 121 The developer of the Security IC Embedded Software must ensure the appropriate usage of Security IC while developing this software in Phase 1 as described in the (i) TOE guidance documents (refer to the Common Criteria assurance class AGD) such as the hardware data sheet, and the hardware application notes, and (ii) findings of the TOE evaluation reports relevant for the Security IC Embedded Software as documented in the certification report.
- 122 Note that particular requirements for the Security IC Embedded Software are often not clear before considering a specific attack scenario during vulnerability analysis of the Security IC (AVA\_VAN). A summary of such results is provided in the document "ETR for composite evaluation" (ETR-COMP). This document will be provided for the evaluation of the composite product. The ETR-COMP may also include guidance for additional tests being required for the combination of hardware and software. The TOE evaluation must be completed before evaluation of the Security IC Embedded Software can be completed. The TOE evaluation can be conducted before and independently from the evaluation of the Security IC Embedded Software.
- 123 The Security IC Embedded Software must ensure the appropriate "Treatment of user data of the Composite TOE (A.Resp-Appl)" as specified below.
- |             |  |
|-------------|--|
| A.Resp-Appl | Treatment of user data of the Composite TOE  |
|             | All user data of the Composite TOE are owned by Security IC Embedded Software. Therefore, it must be assumed that security relevant user data of the Composite TOE (especially cryptographic keys) are treated by the Security IC Embedded Software as defined for its specific application context. |
- 124 The application context specifies how the user data of the Composite TOE shall be handled and protected. The evaluation of the Security IC according to this Security Target is conducted on generalized application context. The concrete requirements for the Security IC Embedded Software shall be defined in the Protection Profile respective Security Target for the Security IC Embedded Software. The Security IC cannot prevent any compromise or modification of user data of the Composite TOE by malicious Security IC Embedded Software.



- 125 The developer of the Security IC Embedded Software must ensure the appropriate “Usage of Key-dependent Functions (A.Key-Function)” while developing this software in Phase 1 as specified below.

A.Key-Function

Usage of Key-dependent Functions

Key-dependent functions (if any) shall be implemented in the Security IC Embedded Software in a way that they are not susceptible to leakage attacks (as described under T.Leak-Inherent and T.Leak-Forced).

- 126 Note that here the routines which may compromise keys when being executed are part of the Security IC Embedded Software. In contrast to this the threats T.Leak-Inherent and T.Leak-Forced address (i) the cryptographic routines which are part of the TOE and (ii) the processing of User Data including cryptographic keys.

# 4 SECURITY OBJECTIVES

127 This chapter4 Security Objectives contains the following sections:

4.1 *Security Objectives for the TOE*

4.2 *Security Objectives for the Security IC Embedded Software*

4.3 *Security Objectives for the Operational Environment*

4.4 *Security Objectives Rationale*

## 4.1 Security Objectives for the TOE

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

- SG1 maintain the integrity 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.

129 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.

130 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 4-1 Standard Security Objectives). Note that the integrity of the TOE is a means to reach these objectives.

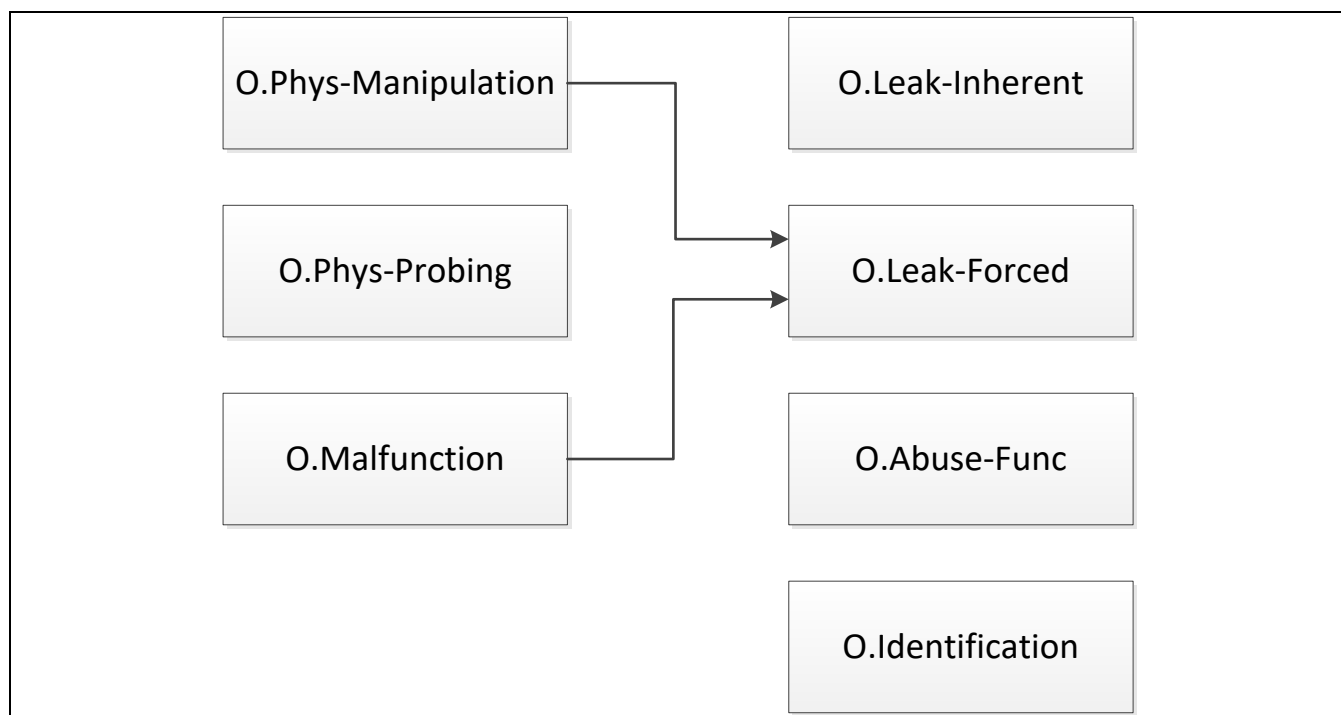


Figure 4-1 Standard Security Objectives

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

- SG4 provide random numbers.

- 132 The additional high-level security considerations are refined below by defining security objectives as required by the Common Criteria (refer to Figure 4-2 Security Objectives related to Specific Functionality).

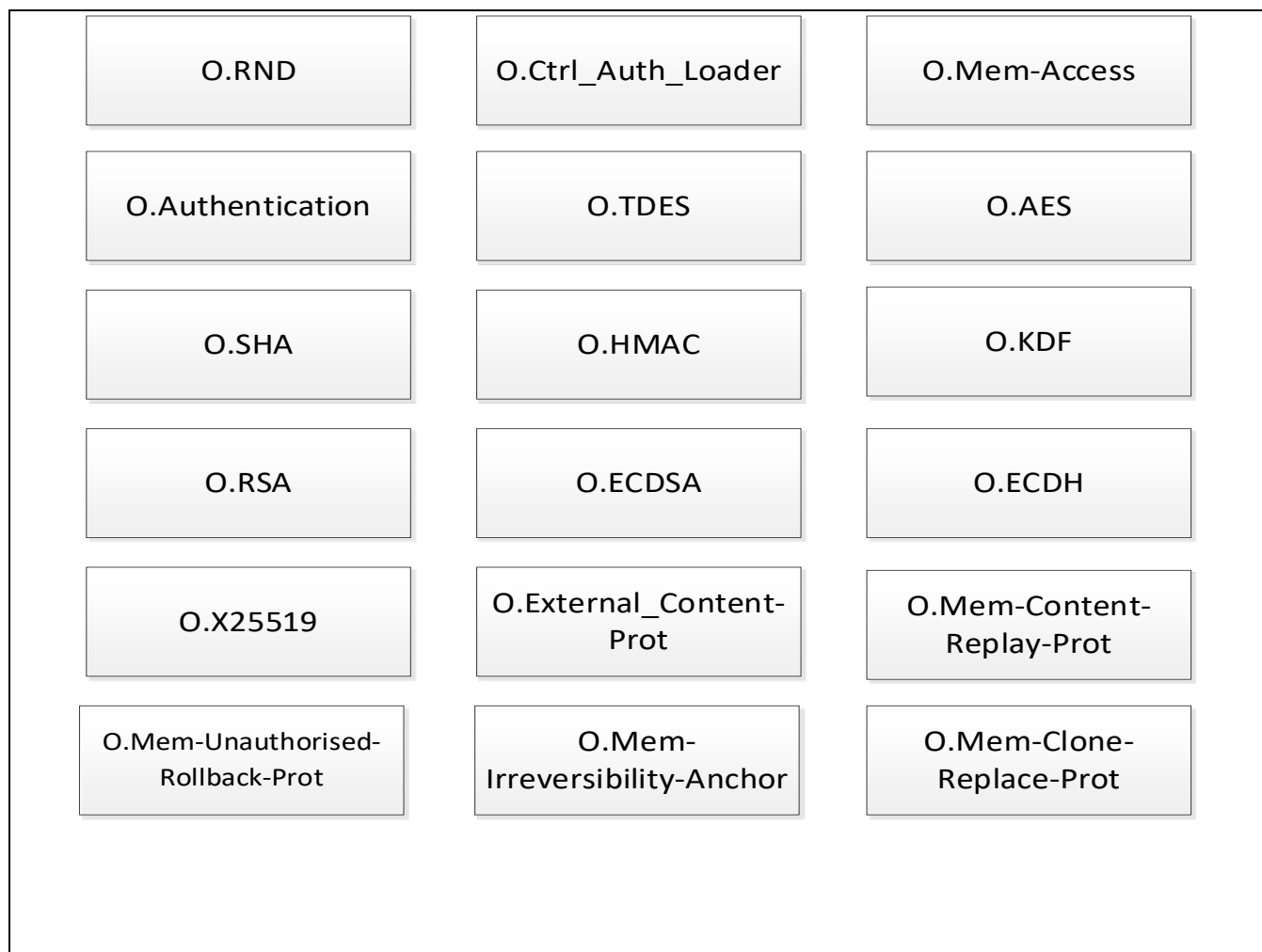


Figure 4-2 Security Objectives related to Specific Functionality

#### 4.1.1 Standard Security Objectives

- 133 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 Security IC

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

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

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

O.Phys-Probing

Protection against Physical Probing

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

This includes protection against

- measuring through galvanic contacts which is direct physical probing on the chips surface except on pads being bonded (using standard tools for measuring voltage and current) or
- measuring not using galvanic contacts but other types of physical interaction between charges (using tools used in solid-state physics research and IC failure analysis)

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

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

135 The TOE shall provide “Protection against Malfunctions (O.Malfunction)” as specified below.

O.Malfunction

Protection against Malfunctions

The TOE must ensure its correct operation.

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

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

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

O.Phys-Manipulation

Protection against Physical Manipulation

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

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

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

- 137 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.

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

O.Abuse-Func                      Protection against Abuse of Functionality

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

- 139 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.

#### 4.1.2 Security Objectives related to Specific Functionality (referring to SG4)

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

O.RND	Random Numbers
	<p>The TOE will ensure the cryptographic quality of random number generation. For instance random numbers shall not be predictable and shall have sufficient entropy.</p> <p>The TOE will ensure that no information about the produced random numbers is available to an attacker since they might be used for instance to generate cryptographic keys.</p>

#### 4.1.3 Security Objectives for Added Function

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

O.Mem-Access	Area based Memory Access Control
	<p>The TOE must provide the Security IC 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.</p>

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

O.Ctrl_Auth Loader	Access control and authenticity for the Loader
	<p>The TSF provides trusted communication channel with authorized user, supports confidentiality protection, replay protection and authentication of the user data to be loaded and access control for usage of the Loader functionality.</p>

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

O.TDES	Cryptographic service Triple-DES
	<p>The TOE provides secure hardware based cryptographic services implementing Triple-DES for encryption and decryption.</p>

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

O.AES	Cryptographic service AES
	<p>The TOE provides secure hardware based cryptographic services for the AES for encryption and decryption.</p>

145 The TOE shall provide “Cryptographic service Hash function (O.SHA)” as specified below.

O.SHA                      Cryptographic service Hash function

The TOE provides secure software and hardware based cryptographic services for secure hash calculation.

146 The TOE shall provide “Cryptographic service Keyed-Hash Message Authentication Code function (O.HMAC)” as specified below.

O.HMAC                      Cryptographic service Keyed-Hash Message Authentication Code function

The TOE provides secure hardware based cryptographic services for secure hash calculation.

147 The TOE shall provide “Cryptographic service Key Derivation Function (O.KDF)” as specified below.

O.KDF                      Cryptographic service Key Derivation Function

The TOE provides secure hardware based cryptographic services for secure AES/HMAC key derivation.

148 The TOE shall provide “Cryptographic service Rivest-Shamir-Adleman (O.RSA)” as specified below.

O.RSA                      Cryptographic service Rivest-Shamir-Adleman

The TOE provides secure software based cryptographic services for Cryptographic operation and Cryptographic key generation.

149 The TOE shall provide “Cryptographic service Elliptic Curve DSA (O.ECDSA)” as specified below.

O.ECDSA                      Cryptographic service Elliptic Curve DSA

The TOE provides secure software based cryptographic services for Cryptographic operation and Cryptographic key generation.

150 The TOE shall provide “Cryptographic service Elliptic Curve Diffie-Hellman (O.ECDH)” as specified below.

O.ECDH                      Cryptographic service Elliptic Curve Diffie-Hellman

The TOE provides secure software based cryptographic services for Cryptographic operation.

151 The TOE shall provide “Cryptographic service Elliptic Curve Diffie-Hellman with curve25519 (O.X25519)” as specified below.

O.X25519                      Cryptographic service Elliptic Curve DH with curve25519

---



The TOE provides secure software based cryptographic services for Cryptographic operation.

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

O. Authentication      Authentication to external entities

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

153 The TOE shall provide “Protection of external Content (O.External-Content-Protect)” as specified below.

O.External-Content-Prot      Protection against disclosure and undetected modification of external memory content.

The content in the external memory must be protected against disclosure and undetected modification, because an attacker can directly access the external memory.

This security objective requires protection of the content stored in external memory by the TOE. The protection prevents disclosure and identifies modifications of stored code and data that is not performed by the TOE.

154 The TOE shall provide “Protection against replay of commands to store or modify data in external memory to the 3S (O.Mem-Command-Replay-Prot)” as specified below.

O.Mem-Command-Replay-Prot      Protection against replay of commands to store or modify data in external memory to the 3S.

The TOE shall protect against replay of content during write, read and erase operations to the external memory by the 3S.

This security objective requires protection against replay of read, write and erase operations. This covers simple replay of previously recorded commands or memory content but also the replay of modified commands or memory content. The TOE shall be able to detect such attacks violating the TOE.

155 The TOE shall provide “Protection against an unauthorised rollback of external memory content (O.Mem-Unauthorized-Rollback-Prot)” as specified below.

O.Mem-Unauthorized-Rollback-Prot      Protection against an unauthorised rollback of external memory content.

The TOE shall protect against replacement of the external memory content with a previous version, even if it was valid in the past.

The security objective requires protection against the simulation of outdated content. Replacement of memory content with a previous version of the same memory content or the manipulations of write operations violate the freshness of the external memory content and shall be detected by the TOE.

- 156 The TOE shall provide “External memory content Irreversibility Anchor (O.Mem-Irreversibility-Anchor)” as specified below.

O.Mem-Irreversibility-Anchor

External memory content Irreversibility Anchor

The TOE shall implement a reference that represents the current content of the external memory. This reference shall be updated based on each authorised modification of the external memory to ensure freshness of the data.

The security objective requires the verification of freshness for data read from the external memory. Therefore, the 3S shall maintain a reference that represents the current content of the external memory. This reference needs to be updated with each authorised read and write operation to detect a violation of the data freshness. It should be maintained in any TOE operational state, including the standby and sleep states. In the case of non-volatile memory, the Irreversibility Anchor needs to be persistently saved between two boots.

- 157 The TOE shall provide “Protection against external memory cloning or replacement (O.Mem-Clone-Replace-Prot)” as specified below.

O.Mem-Clone-Replace-Prot

Protection against external memory cloning or replacement.

The TOE shall protect against cloning or replacement of content with the content stored in the memory of another instance of the TOE and against replacement of the external memory with the one from another instance of the TOE.

The security objective requires protection against replacement of its external memory content with the content of another instance of the TOE. The external memory content shall only be valid for the 3S that is initially linked to this external memory. The replacement of the external memory or the transfer of the content from a memory unit that is linked to another instance of the TOE shall be detected.

## 4.2 Security Objectives for the Security IC Embedded Software

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

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

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

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

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

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

### 4.2.1 Clarification of “Treatment of User Data of the Composite TOE(OE.Resp-Appl)”

- 160 Regarding the cryptographic services this objective of the environment has to be clarified. By definition cipher or plain text data and cryptographic keys are User Data. The Security IC Embedded Software shall treat these data appropriately, use only proper secret keys (chosen from a large key space) as input for the cryptographic function of the TOE and use keys and functions appropriately in order to ensure the strength of cryptographic operation.
- 161 This means that keys are treated as confidential as soon as they are generated. The keys must be unique with a very high probability, as well as cryptographically strong. If keys are imported into the TOE and/or derived from other keys, quality and confidentiality must be maintained. This implies that appropriate key management has to be realised in the environment.
- 162 Regarding the area based access control this objective of the environment has to be clarified. The treatment of User Data of the Composite TOE is also required when a multi-application operating system is implemented as part of the Security IC Embedded Software on the TOE. In this case the multi-application operating system should not disclose security relevant user data of one application to another application when it is processed or stored on the TOE.

### 4.3 Security Objectives for the Operational Environment

163 TOE Delivery up to the End of Phase 6

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

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

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

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

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

OE.Loader\_Usage      Secure communication and usage of the Loader

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

166 The operational environment shall provide “External entities authenticating of the TOE (OE.TOE\_Auth)”.

OE.TOE\_Auth      External entities authenticating the TOE

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

#### 4.3.1 Clarification of “Protection during Composite Product Manufacturing (OE.Process-Sec-IC)”

167 The protection during finishing and personalization includes also the personalization process and the personalization data during Phase 5 and Phase 6.

168 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.

#### 4.4 Security Objectives Rationale

169 Table 4-1 below gives an overview, how the assumptions, threats, and organisational security policies are addressed by the objectives. The text following after the table justifies this in detail.

Table 4-1 Security Objectives versus Assumptions, Threats or Policies

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.RND	O.RND	
T.Mem-Access	O.Mem-Access	
P.Crypto-Service	O.TDES O.AES O.RSA O.ECDSA O.ECDH O.X25519 O.SHA O.HMAC O.KDF	
A.Key-Function	OE.Resp-Appl	
P.Ctrl_Loader	O.Ctrl_Auth_Loader OE.Loader_Usage	
T.Masquerade_TOE	O.Authentication OE.TOE_Auth	
T.External-Content-Abuse	O.External-Content-Protection	
T.Mem-Command-Replay	O.Mem-Command-Replay-Protection O.Mem-Irreversibility-Anchor	
T.Mem-Unauthorised-Rollback	O.Mem-Irreversibility-Anchor O.Mem-Unauthorised-Rollback-Protection	
T.Mem-Clone-Replace	O.Mem-Clone-Replace-Protection	

- 170 The justification related to the assumption “Treatment of user data of the Composite TOE (A.Resp-Appl)” is as follows:
- 171 Since OE.Resp-Appl requires the Security IC Embedded Software to implement measures as assumed in A.Resp-Appl, the assumption is covered by the objective.
- 172 The justification related to the organisational security policy “Protection during TOE Development and Production (P.Process-TOE)” is as follows:
- 173 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 78. 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.
- 174 The justification related to the assumption “Protection during Packaging, Finishing and Personalisation (A.Process-Sec-IC)” is as follows:
- 175 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.
- 176 The justification related to the threats “Inherent Information Leakage (T.Leak-Inherent)”, “Physical Probing (T.Phys-Probing)”, “Malfunction due to Environmental Stress (T.Malfunction)”, “Physical Manipulation (T.Phys-Manipulation)”, “Forced Information Leakage (T.Leak-Forced)”, “Abuse of Functionality (T.Abuse-Func)” and “Deficiency of Random Numbers (T.RND)” is as follows:
- 177 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.
- 178 The justification related to the threat “Memory Access Violation (T.Mem-Access)” is as follows:
- 179 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 Security IC 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.
- 180 The clarification of O.Mem-Access makes clear that it is up to the Security IC 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 Security IC Embedded Software. This is further emphasised by the clarification of Treatment of User Data of the Composite TOE(OE.Resp-Appl) which reminds that the Security IC Embedded Software must not undermine the restrictions it defines. Therefore, the clarifications contribute to the coverage of the threat T.Mem-Access. .
- 181 Compared to Security IC Platform Protection Profile a clarification has been made for the security objective “Treatment of User Data of the Composite TOE(OE.Resp-Appl)”: By definition cipher or plain text data and cryptographic keys are User Data. So, the Security IC Embedded Software will protect such data if required and use keys and functions appropriately in order to ensure the strength of cryptographic operation. Quality and confidentiality must be maintained for keys that are imported and/or derived from other keys. That is
-

expressed by the assumption A.Key—Function which is covered from OE.Resp—Appl. These measures make sure that the assumption A.Resp-Appl is still covered by the security objective OE.Resp-Appl.

- 182 The organisational security policy “Controlled usage to Loader Functionality (P.Ctrl\_Loader) is directly implemented by the security objective for the TOE “Access control and authenticity for the Loader (O.Ctrl\_Auth\_Loader)” and the security objective for the TOE environment “Secure communication and usage of the Loader (OE.Loader\_Usage)”.
- 183 The threat “Masquerade the TOE (T.Masquerade\_TOE)” is directly covered by the TOE security objective “Authentication to external entities (O.Authentication)” describing the proving part of the authentication and the security objective for the operational environment of the TOE “External entities authenticating of the TOE (OE.TOE\_Auth)” the verifying part of the authentication.
- 184 The justification related to the security objectives O.TDES, O.AES, O.RSA, O.ECDSA, O.ECDH, O.X25519, O.SHA, O.HMAC and O.KDF is followings: Since these objectives require the TOE to implement the same specific security functionality as required by P.Crypto-Service, the organization security policy is covered by the objective.
- 185 T.External-Content-Abuse is countered by O.External-Content-Protection, which requires the TOE to prevent disclosure and undetected modification of the content stored in external memory.
- 186 T.Mem-Command-Replay is countered by O.Mem-Command-Replay-Prot and O.Mem-Irreversibility-Anchor as follows:
- O.Mem-Command-Replay-Prot requires protection against replay of commands exported from the 3S in the external NVM mitigating T.Mem-Command-Replay.
  - O.Mem-Irreversibility-Anchor requires the implementation of a reference inside the 3S representing the current content of the external memory. The reference inside the 3S is updated associated with each change issued by the 3S on the external memory. This reference allows to verify the freshness of the data when they are loaded from the external memory.
- 187 T.Mem-Unauthorised-Rollback is countered by O.Mem-Unauthorized-Rollback-Prot and O.Mem-Irreversibility-Anchor as follows:
- O.Mem-Unauthorized-Rollback-Prot requires that the TOE protects against replacement of external memory content with older content of the same external memory, where the data freshness property is not met, thus, mitigating this threat.
  - O.Mem-Irreversibility-Anchor requires that the TOE implements a reference inside the 3S representing the current content of the external memory. The reference inside the 3S is updated associated with each change issued by the 3S on the external memory. This reference allows to verify the freshness of the data when they are loaded from the external memory.
- 188 T.Mem-Clone-Replace is countered by O.Mem-Clone-Replace-Protection, which requires the TOE to detect the replacement of the external memory content with one of a different TOE’s memory, or physical replacement of the external memory with the external memory of a different instance of the TOE.



# 5 EXTENDED COMPONENTS DEFINITION

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

5.1 Definition of the family FCS\_RNG

5.2 Definition of the Family FMT\_LIM

5.3 Definition of the Family FAU\_SAS

5.4 Definition of the Family FDP\_SDC

5.5 Definition of the Family FIA\_API

0 Definition of the Family FDP\_URC

5.7 Definition of the Family FDP\_IRA

## 5.1 Definition of the Family FCS\_RNG

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

FCS\_RNG Generation of Random Numbers

Family behaviour

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

Component levelling:



FCS_RNG.1	Generation of random numbers requires that random numbers meet a defined quality metric.
Management:	FCS_RNG.1 There are no management activities foreseen.
Audit:	FCS_RNG.1 There are no actions defined to be auditable.
FCS_RNG.1	Random number generation
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FCS_RNG.1.1	The TSF shall provide a [selection: <i>physical, non-physical true, deterministic, hybrid physical, hybrid deterministic</i> ] random number generator that implements: [assignment: <i>list of security capabilities</i> ].
FCS_RNG.1.2	The TSF shall provide [selection: <i>bits, octets of bits, numbers</i> ] [assignment: <i>format of the numbers</i> ] that meet [assignment: <i>a defined quality metric</i> ].

## 5.2 Definition of the Family FMT\_LIM

191 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.

192 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.

193 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 policy*].

Dependencies: FMT\_LIM.2 Limited availability.

194 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 its availability so that in conjunction with "Limited capabilities (FMT\_LIM.1)" the following policy is enforced [assignment: *Limited availability policy*].

Dependencies: FMT\_LIM.1 Limited capabilities.

Application note: The functional requirements FMT\_LIM.1 and FMT\_LIM.2 assume that there are two types of mechanisms (limitation of capabilities and limitation of availability) which together shall provide protection in order to enforce the same policy or two mutual supportive policies related to the same functionality. This allows e.g. that

(i) the TSF is provided without restrictions in the product in its user environment but its capabilities are so limited that the policy is enforced

or conversely

(ii) the TSF is designed with high functionality but is removed or disabled in the product in its user environment.

### 5.3 Definition of the Family FAU\_SAS

195 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.

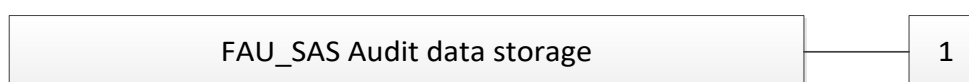
196 The family "Audit data storage (FAU\_SAS)" is specified as follows.

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 [selection: *the Initialisation Dta, Pre-Personalisation Data*, [assignment: *list of audit information*]] in the [assignment: *type of persistent memory*].

Dependencies: No dependencies.

## 5.4 Definition of the Family FDP\_SDC

197 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.

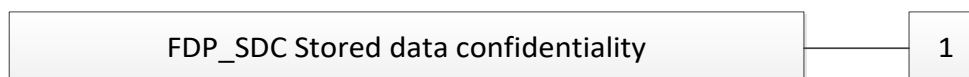
198 The family “Stored data confidentiality (FDP\_SDC)” is specified as follows.

FDP\_SDC.1 Stored data confidentiality

Family behaviour

This family provides requirements that address protection of user data confidentiality while these data are stored within memory areas protected by the TSF. The TSF provides access to the data in the memory through the specified interfaces only and prevents compromise of their information bypassing these interfaces. It complements the family “Stored data integrity (FDP\_SDI)” which protects the user data from integrity errors while being stored in the memory.

Component leveling



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]

## 5.5 Definition of the Family FIA\_API

- 199 To describe the IT security functional requirements of the TOE a functional family FIA\_API (Authentication Proof of Identity) of the Class FIA (Identification and authentication) is defined here. This family describes the functional requirements for the proof of the claimed identity by the TOE and enables the authentication verification by an external entity. The other families of the class FIA address the verification of the identity of an external entity by the TOE.
- 200 The other families of the Class FIA describe only the authentication verification of users' identity performed by the TOE and do not describe the functionality of the user to prove their identity. The following paragraph defines the family FIA\_API in the style of the Common Criteria part 2 (cf. [3], chapter "Extended components definition (APE\_ECD)") from a TOE point of view.
- 201 The family "Authentication Proof of Identity (FIA\_API)" is specified as follows.

FIA\_API.1 Authentication Proof of Identity

Family behaviour

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

Component levelling



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

Management: FIA\_API.1

The following actions could be considered for the management functions in FMT: Management of authentication information used to prove the claimed identity.

Audit: FIA\_API.1

There are no actions defined to be auditable.

FIA\_API.1 Authentication Proof of Identity

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA\_API.1.1 The TSF shall provide a [assignment: authentication mechanism] to prove the identity of the [selection: TOE, [assignment: object, authorized user or role]] to an external entity.

## 5.6 Definition of the Family FDP\_URC

202 To define security requirements of the TOE the additional family (FDP\_IRA) of the Class FDP (User data protection) to verify the freshness of data stored in a physically separated storage is defined here. This family defines mechanisms to determine whether the content read from a physically separated memory meets the property of data freshness, by verifying that they are those resulting from the latest authorized operation (write or erase) of the TSF that modifies the content in the physically separated memory. If the content read from the physically separated memory cannot be uniquely linked to the latest write or erase operation executed by the TSF, the data freshness property is not met, and the read data is rejected.

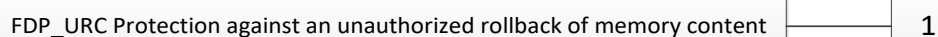
203 The family “Protection against an unauthorized rollback of stored contents (FDP\_URC)” is specified as follows.

FIA\_URC                      Protection against an unauthorized rollback of memory content

Family behaviour

This family defines functional requirements for the detection of an unauthorized rollback of content stored in the external memory.

Component levelling



FDP\_URC.1                      Requires the TOE to protect against an unauthorized rollback of the content stored in the external memory.

Management:                      FDP\_URC.1

There are no actions defined to be auditable.

Audit:                              FDP\_URC.1

There are no actions defined to be auditable.

FDP\_URC.1                      Protection against an unauthorized rollback of memory content

Hierarchical to:                      No other components.

Dependencies:                      No dependencies.

FDP\_URC.1.1                      The TOE shall detect an unauthorized replacement of the content stored in *[assignment: physically separated memory]* before the content is used. The detection shall be effective in any case where modification or read operation depends on the current content of this external memory.

FDP\_URC.1.2                      Upon detection of unauthorized rollback of the content stored in a physically separated memory, the TOE shall [selection: stop TOE operation, *[assignment: other actions]]*



## 5.7 Definition of the Family FDP\_IRA

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

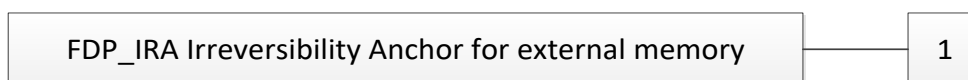
205 The family “Irreversibility Anchor for external memory (FDP\_IRA)” is specified as follows.

FDP\_IRA Irreversibility Anchor for external memory

Family behaviour

This family provides requirements for the implementation of a mechanism that verifies that read operations from this physically separated memory represent always the latest authorized modification of this memory. The TSF provides an irreversibility anchor maintaining a link between a transaction counter associated write or erase operation and the data transferred to a physically separated memory. Thereby, the irreversibility anchor allows to determine, if a data read operation from the physically separated memory represents the data based on the latest write or erase operation. The anchor is implemented in an irreversible way representing unique states, i.e., without the possibility of going back to previous states. The pattern maintained by the irreversibility anchor value allows to verify the data freshness provided by subsequent read operations to the physically separated memory. If the physically separated memory is a non-volatile memory, the irreversibility anchor shall be maintained in any operational state of the TOE.

Component levelling



FDP\_IRA.1 Requires the TOE to verify that read operations from a physically separated memory represent always the latest authorized modification of this memory.

Management: There are no management activities foreseen.

Audit: The following actions should be auditable if FAU\_GEN Security audit data generation is included in the PP/ST:

- Any violation of the data freshness detected upon a read operation from the physically separated memory.

FDP\_IRA.1 Irreversibility Anchor for external memory

Hierarchical to: No other components.

Dependencies: No dependencies.

FDP\_IRA.1.1 The TSF shall verify the freshness of data for each read operation from the *[assignment: physically separated memory]*.

FDP\_IRA.1.2 The irreversibility anchor shall maintain a distinct transaction references for each *[selection: write, erase, [assignment: further operation that changes the content of the physically separated memory]]* operation and that is unambiguously linked with the current content of the transaction with the associated physically separated memory.

FDP\_IAR.1.3

The state of the irreversibility anchor implemented by the TSF shall be maintained during *[selection: operation, power off, power saving, any operation mode]*.

# 6 IT security requirements

206 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

- 207 In order to define the Security Functional Requirements the Part 2 of Common Criteria and the Protection Profile [5] was used.
- 208 However, some Security Functional Requirements have been refined. The refinements are described by the associated SFRs below.
- 209 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.
  - Iteration is denoted by showing a slash “/”.

### 6.1.1 Malfunctions

- 210 The TOE shall meet the requirement “Limited fault tolerance (FRU\_FLT.2)” as specified below.
- |                   |  |
|-------------------|--|
| FRU_FLT.2         | Limited fault tolerance  |
| Hierarchical to:  | FRU_FLT.1 Degraded fault tolerance   |
| FRU_FLT.2.1       | The TSF shall ensure the operation of all the TOE’s capabilities when the following failures occur: <i>exposure to operating conditions which are not detected according to the requirement Failure with preservation of secure state (FPT_FLS.1).</i> |
| 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.   |
| Application Note: | Environmental conditions include but are not limited to power supply, clock, and other external signals (e.g. reset signal) necessary for the TOE operation.   |
- 211 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: <i>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.</i> |
| Dependencies:    | No dependencies  |
| Refinement:      | The term “failure” above also covers “circumstances”. The TOE prevents failures for the “circumstances” defined above.   |

### 6.1.2 Abuse of Functionality

212 The TOE shall meet the requirement “Limited capabilities (FMT\_LIM.1)” as specified below (Common Criteria Part 2 extended).

FMT\_LIM.1/Test Limited capabilities

Hierarchical to: No other components.

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

Dependencies: FMT\_LIM.2/Test Limited availability.

FMT\_LIM.1/Debug Limited capabilities

Hierarchical to: No other components.

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

Dependencies: FMT\_LIM.2/Debug Limited availability.

213 The TOE shall meet the requirement “Limited availability (FMT\_LIM.2)” as specified below (Common Criteria Part 2 extended).

FMT\_LIM.2/Test Limited availability

Hierarchical to: No other components.

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

Dependencies: FMT\_LIM.1/Test Limited capabilities.

FMT\_LIM.2/Debug Limited availability

Hierarchical to: No other components.

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

Dependencies: FMT\_LIM.1/Debug Limited capabilities.

214 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 Prepersonalisation Data in an *OTP*.

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.

### 6.1.3 Physical Manipulation and Probing

215 The TOE shall meet the requirement “Stored data confidentiality (FDP\_SDC.1)” as specified below (Common Criteria Part 2 extended).

FDP\_SDC.1 Stored data confidentiality

Hierarchical to: No other components.

Dependencies: No dependencies.

FDP\_SDC.1.1 The TSF shall ensure the confidentiality of the information of the user data while it is stored in the *RAM or ROM*.

Refinement: The asset “user data” selected above has been refined to include “user data” and “TSF data”. The “user data” is applicable for data stored in RAM while “TSF data” is applicable for data stored in both RAM or ROM.

216 The TOE shall meet the requirement “Stored data integrity monitoring and action (FDP\_SDI.2)” as specified below.

FDP\_SDI.2 Stored data integrity monitoring and action

Hierarchical to: FDP\_SDI.1 Stored data integrity monitoring

Dependencies:	No dependencies.
FDP_SDI.2.1	The TSF shall monitor user data stored in containers controlled by the TSF for ECC error or <i>Parity error</i> on all objects, based on the following attributes: <i>RAM or ROM read operation</i> .
Refinement:	The asset “user data” selected above has been refined to include “user data” and “TSF data”. The “user data” is applicable for data stored in RAM while “TSF data” is applicable for data stored in both RAM or ROM.
FDP_SDI.2.2	Upon detection of a data integrity error, the TSF shall <i>enforce a device an interrupt (IRQ)</i> .
Application Note:	This requirement is achieved by security features such as memory encryption, bus scrambling, security detectors and memory access control.

217 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 <i>physical manipulation and physical probing</i> 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 a IRQ occurs to stops operation if a physical manipulation or physical probing attack is detected. And also Static Address/Data scrambling for bus and memory & Synthesizable processor core 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.

#### 6.1.4 Leakage

218 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 <i>Data Processing Policy</i> to prevent the <u>disclosure</u> 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.

219 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 <u>disclosure</u> 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.

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

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

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

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

### 6.1.5 Random Numbers (DTRNG/DRBG)

222 The TOE shall meet the requirement “Quality metric for random numbers (FCS\_RNG.1)” as specified below (Common Criteria Part 2 extended).

**FCS\_RNG.1/PTG.2      Random number generation - PTG.2**



Hierarchical to:	No other components.
FCS_RNG.1.1/PTG.2	The TSF shall provide a <u>physical</u> true random number generator that implements:
(PTG.2.1)	<i>A total failure test detects a total failure of entropy source immediately when the RNG has started. When a total failure is detected, no random numbers will be output.</i>
(PTG.2.2)	<i>If a total failure of the entropy source occurs while the RNG is being operated, the RNG <u>prevents the output of any internal random number that depends on some raw random numbers that have been generated after the total failure of the entropy source.</u></i>
(PTG.2.3)	<i>The online test shall detect non-tolerable statistical defects of the raw random number sequence (i) immediately when the RNG has started, and (ii) while the RNG is being operated. The TSF must not output any random numbers before the power-up online test has finished successfully or when a defect has been detected.</i>
(PTG.2.4)	<i>The online test procedure shall be effective to detect non-tolerable weaknesses of the random numbers soon.</i>
(PTG.2.5)	<i>The online test procedure checks the quality of the raw random number sequence. It is triggered, <u>externally applied upon specified internal events (i.e., crypto key generation).</u> The online test is suitable for detecting non-tolerable statistical defects of the statistical properties of the raw random numbers within an acceptable period of time.</i>
FCS_RNG.1.2/PTG.2	The TSF shall provide <u>numbers, 32-bit per number</u> that meet:
(PTG.2.6)	<i>Test procedure A does not distinguish the internal random numbers from output sequences of an ideal RNG.</i>
(PTG.2.7)	<i>The average Shannon entropy per internal random bit exceeds 0.997.</i>
Application Note:	The DTRNG library comprises some functions that perform statistical tests on the DTRNG output in order to execute so-called total-failure test and online test. The online test function triggers a set of statistical tests embedded in a logic block connecting to DTRNG hardware directly, Upon completing the statistical tests, the logic block shall notify embedded software of the test result. The total-failure test is implemented in pure software. If either test fails, the function returns an error value and the DTRNG is shut down. Those functions are described in DTRNG Application note in detail and are available to embedded software.
Dependencies:	No dependencies

### 223 FCS\_RNG.1/DRG.3 Random number generation - DRG.3

Hierarchical to: No other components.

FCS\_RNG.1.1/DRG.3 The TSF shall provide a deterministic random number generator that implements:

(DRG.3.1) *If initialized with a random seed using a PTRNG of class PTG.2 as random*

source, the internal state of the RNG shall have at least 100-bits of entropy.

(DRG.3.2) The RNG provides forward secrecy.

(DRG.3.3) The RNG provides backward secrecy even if the current internal state is known.

FCS\_RNG.1.2/DRG.3 The TSF shall provide random numbers that meet:

(DRG.3.4) The RNG, initialized with a random seed after operating the start test and on-line test, generates output for which  $2^{35}$  strings of bit length 128 are mutually different with probability  $(1-2^{-17})$ .

(DRG.3.5) Statistical test suites cannot practically distinguish the random numbers from output sequences of an ideal RNG. The random numbers must pass test procedure A.

Dependencies: No dependencies

### 6.1.6 Memory Access Control

- 224 Usage of multiple applications in one Security IC often requires separating code and data in order to prevent that one application can access code and/or data of another application. To support the TOE provides Area based Memory Access Control.
- 225 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 where it applies. The security functional requirement "Security attribute based access control (FDP\_ACF.1)" defines addresses 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 and memory areas. The corresponding permission control information is evaluated "on-the-fly" by the hardware so that access is granted/effective or denied/inoperable.
- 226 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).
- 227 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.
- 228 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 read, write, delete, and execute accesses of software running at between two different modes (privilege and user mode) on data including code stored in memory areas.

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 privilege mode).

229 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 all subjects (software with privilege mode and user mode), all objects (data including code stored in memories) and all the operations defined in the Memory Access Control Policy.*

*Subjects are software codes in Privilege and User mode.*

*Objects are data stored in ROM, RAM and OTP memories.*

Dependencies: FDP\_ACF.1 Security attribute based access control

230 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 all the operations related to the data stored in memories, which are *the read, write and 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 following: *memory area where the software is executed from and/or the memory area where the access is performed to and/or the operation to be performed.*

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

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

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

Dependencies: FDP\_ACC.1 Subset access control  
FMT\_MSA.3 Static attribute initialisation

231 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 <i>Memory Access Control Policy</i> to provide <i>well defined</i> default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2	The TSF shall allow <i>any subject (provided that the Memory Access Control Policy is enforced and the necessary access is therefore allowed)</i> 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
232	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 <i>Memory Access Control Policy</i> to restrict the ability to <u>change default, modify or delete</u> the security attributes <i>permission control information to running at privilege mode</i> .
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
233	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 management functions: <i>access the control registers of the MPU</i> .
Dependencies:	No dependencies

### 6.1.7 Cryptographic Support

- 234 FCS\_COP.1 Cryptographic operation requires, a cryptographic operation to be performed in accordance with a specified algorithm and with a cryptographic key of specified sizes. The specified algorithm and cryptographic key sizes can be based on an assigned standard.
- 235 The following additional specific security functionality is implemented in the TOE:
- Triple Data Encryption Standard (TDES) with 112bit or 168bit key size
  - Advanced Encryption Standard (AES) with 128 bit, 192bit and 256bit key size
  - Rivest-Shamir-Adleman (RSA) public key asymmetric cryptography, with key size 1900-bit up to 4096-bit with a granularity of 2 bits (optional)
  - Elliptic Curve Cryptography (ECC) (optional)
  - Secure Hash Algorithm (SHA) (optional)

### 6.1.8 Triple-DES Operation

236 The Triple DES (TDES) operation of the TOE shall meet the requirement “Cryptographic operation (FCS\_COP.1)” as specified below.

FCS_COP.1/TDES	Cryptographic operation – TDES
Hierarchical to:	No other components.
FCS_COP.1.1/TDES	The TSF shall perform <i>encryption and decryption</i> in accordance with a specified cryptographic algorithm <i>Triple Data Encryption Standard (TDES) – ECB mode</i> and cryptographic key sizes <i>112 bit or 168 bit key size</i> that meet the following: <i>[FIPS SP800-67], chapter 2 and 3. TOE implements TDES with key option 1 and 2 with ECB mode.</i>
Dependencies:	[FDP_ITC.1 Import of user data without security attributes or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction

### 6.1.9 AES Operation

237 The AES operation of the TOE shall meet the requirement “Cryptographic operation (FCS\_COP.1)” as specified below.

FCS_COP.1/AES	Cryptographic operation – AES
Hierarchical to:	No other components.
FCS_COP.1.1/AES	The TSF shall perform <i>encryption and decryption</i> in accordance with a specified cryptographic algorithm <i>Advanced Encryption Standard (AES) – ECB, CTR, CBC and GCM mode</i> and cryptographic key sizes <i>128bit, 192bit or 256bit key size</i> that meet the following standard: <i>[FIPS197], chapter 5.</i>
Dependencies:	[FDP_ITC.1 Import of user data without security attributes or FDP_ITC.2 Import of user data with security attributes or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction

### 6.1.10 Key Manager (KDF) Operation

238 The Key Manager (KDF) operation of the TOE shall meet the requirement “Cryptographic operation (FCS\_COP.1)” as specified below.

FCS_COP.1/KDF_KeyWrap	Cryptographic operation – KDF with Key Wrap
Hierarchical to:	No other components.
FCS_COP.1.1/KDF_KeyWrap	The TSF shall perform <i>key management of AES</i> in accordance with a specified key derivation function with <i>Key wrap (AES) and cryptographic</i>

key sizes *128bit, 192bit or 256bit* key size that meet the following standard: [SP800-38F].

Dependencies: [FDP\_ITC.1 Import of user data without security attributes or FDP\_ITC.2 Import of user data with security attributes or FCS\_CKM.1 Cryptographic key generation] FCS\_CKM.4 Cryptographic key destruction

239 The Key Manager (KDF) operation of the TOE shall meet the requirement “Cryptographic operation (FCS\_COP.1)” as specified below.

FCS\_COP.1/KDF\_KBKDF Cryptographic operation – KDF with KBKDF

Hierarchical to: No other components.

FCS\_COP.1.1/KDF\_KBKDF The TSF shall perform *key management of HMAC* in accordance with a specified key derivation function with *KBKDF (HMAC) in Counter mode - SHA2-256/384/512, SHA3-224/256/384/512 and cryptographic key sizes 256bit, 512bit, 768bit, 1024bit, 1152bit* that meet the following standard: [SP800-108].

Dependencies: [FDP\_ITC.1 Import of user data without security attributes or FDP\_ITC.2 Import of user data with security attributes or FCS\_CKM.1 Cryptographic key generation] FCS\_CKM.4 Cryptographic key destruction

### 6.1.11 Secure Hash Algorithm (SHA)

240 This SFR related to SHA-2/ SHA-3/ HMAC hardware engines in the Security Controller.

241 The Secure Hash Algorithm (SHA) of the TOE shall meet the requirement “Cryptographic operation (FCS\_COP.1)” as specified below.

FCS\_COP.1/SHA\_HW Cryptographic operation-SHA

Hierarchical to: No other components.

FCS\_COP.1.1/SHA\_HW The TSF shall perform *secure hash computation* in accordance with a specified cryptographic algorithm *SHA2-256/384/512, SHA3-224/256/384/512, SHAKE128/256* and cryptographic key sizes *none* that meet the following standard: [FIPS PUB 180-3], [FIPS PUB 202]

Dependencies: [FDP\_ITC.1 Import of user data without security attributes or FDP\_ITC.2 Import of user data with security attributes, or FCS\_CKM.1 Cryptographic key generation] FCS\_CKM.4 Cryptographic key destruction

### 6.1.12 Hash-based Message Authentication Code (HMAC)

242 The Hash-based Message Authentication Code (HMAC) of the TOE shall meet the requirement “Cryptographic operation (FCS\_COP.1)” as specified below.

FCS\_COP.1/HMAC Cryptographic operation-HMAC

Hierarchical to: No other components.

FCS\_COP.1.1/HMAC The TSF shall perform *keyed-Hash Message Authentication Code* in accordance with a specified cryptographic algorithm *HMAC* and cryptographic key sizes *SHA2-256/384/512, SHA3-224/256/384/512* that meet the following standard: *[FIPS PUB 198-1]*

Dependencies: [FDP\_ITC.1 Import of user data without security attributes or  
FDP\_ITC.2 Import of user data with security attributes, or  
FCS\_CKM.1 Cryptographic key generation]  
FCS\_CKM.4 Cryptographic key destruction

### 6.1.13 Rivest-Shamir-Adleman (RSA) Operation (optional)

243 The AH3 Secure RSA/ECC/SHA library of the TOE shall meet the requirement “Cryptographic operation (FCS\_COP.1)” as specified below.

FCS\_COP.1/RSA Cryptographic operation – RSA

Hierarchical to: No other components.

FCS\_COP.1.1/RSA The TSF shall perform *the modular exponentiation part of RSA signature generation and verification* in accordance with a specified cryptographic algorithm Rivest-Shamir-Adleman (RSA:standard RSA and RSA-CRT) and cryptographic key sizes *from 1900-bit up to 4096-bit with 2-bit granularity* that meet the following standard: *[ISO/IEC14888-2:2008]] section 6.2 and 6.3.*

Dependencies: [FDP\_ITC.1 Import of user data without security attributes or  
FDP\_ITC.2 Import of user data with security attributes or  
FCS\_CKM.1 Cryptographic key generation]  
FCS\_CKM.4 Cryptographic key destruction

### 6.1.14 Rivest-Shamir-Adleman (RSA) Operation (optional)

244 The RSA key generation for the AH3 Secure RSA/ECC/SHA library shall meet the requirement “Cryptographic key generation (FCS\_CKM.1)” as specified below.

FCS\_CKM.1/RSA Cryptographic key generation – RSA

Hierarchical to: No other components.

FCS\_CKM.1.1/RSA The TSF shall generate cryptographic keys in accordance with the specified cryptographic key generation algorithm *RSA* and with the specified cryptographic key sizes *from 1900-bit up to 4096-bit with 2-bit granularity* that meet the following standard: *[ETSI TS 102 176-1], section 6.2.2.1 Key and parameter generation algorithm rsagen1 and [ISO 18032], Incremental search.*

Dependencies: [FCS\_CKM.2 Cryptographic key distribution or  
FCS\_COP.1 Cryptographic operation]  
FCS\_CKM.4 Cryptographic key destruction

### 6.1.15 Elliptic Curve DSA Operation (optional)

245 The ECC library of the TOE shall meet the requirement “Cryptographic operation (FCS\_COP.1)” as specified below.

FCS\_COP.1/ECDSA Cryptographic operation – ECDSA

Hierarchical to: No other components.

FCS\_COP.1.1/ECDSA The TSF shall perform *the signature generation/verification* in accordance with the specified cryptographic algorithm *ECDSA* and cryptographic key sizes *from 224-bit up to 512-bit* that meet the following standard: [ANS X9.62], section 7.3 *Signing Process* and section 7.4 *Verifying Process*.

Dependencies: [FDP\_ITC.1 Import of user data without security attributes or  
FDP\_ITC.2 Import of user data with security attributes, or  
FCS\_CKM.1 Cryptographic key generation]  
FCS\_CKM.4 Cryptographic key destruction

Note 1: The AH3 Secure RSA/ECC/SHA library supports any valid curves over prime fields of size from 224-bit to 512-bit. However standard curves listed below whose security has been proven are in the scope of this evaluation. 1) [NIST curves]: Curves P-224, P-256, P-384 2) [Brainpool curves]: brainpoolP224r1, brainpoolP224t1, brainpoolP256r1, brainpoolP256t1, brainpoolP320r1, brainpoolP320t1, brainpoolP384r1, brainpoolP384t1, brainpoolP512r1, brainpoolP512t1, 3) [SEC-recommended curves]: secp224k1, secp224r1, secp256k1, secp256r1, secp384r1

### 6.1.16 Elliptic Curve DSA Key Generation (optional)

246 The key generation for the ECC library shall meet the requirement “Cryptographic key generation (FCS\_CKM.1)” as specified below.

FCS\_CKM.1/ECDSA Cryptographic key generation – ECDSA

Hierarchical to: No other components.

FCS\_CKM.1.1/ECDSA The TSF shall generate cryptographic keys in accordance with the cryptographic key generation algorithm *ECC* and with the cryptographic key sizes *from 224-bit up to 512-bit* that meet the following standard: [ANS X9.62], section A.4.3 *Elliptic Curve Key Generation*.

Dependencies: [FCS\_CKM.2 Cryptographic key distribution or  
FCS\_COP.1 Cryptographic operation]  
FCS\_CKM.4 Cryptographic key destruction



Note 1: The AH3 Secure RSA/ECC/SHA library supports any valid curves over prime fields of size from 224-bit to 512-bit. However standard curves listed below whose security has been proven are in the scope of this evaluation. 1) [NIST curves]: Curves P-224, P-256, P-384, P-521 2) [Brainpool curves]: brainpoolP224r1, brainpoolP224t1, brainpoolP256r1, brainpoolP256t1, brainpoolP320r1, brainpoolP320t1, brainpoolP384r1, brainpoolP384t1, brainpoolP512r1, brainpoolP512t1, 3) [SEC-recommended curves]: secp224k1, secp224r1, secp256k1, secp256r1, secp384r1, secp521r1

### 6.1.17 Elliptic Curve Diffie-Hellman (ECDH) Key Agreement (optional)

247 The ECC library of the TOE shall meet the requirement “Cryptographic operation (FCS\_COP.1)” as specified below.

FCS\_COP.1/ECDH Cryptographic operation – ECDH

Hierarchical to: No other components.

FCS\_COP.1.1/ECDH The TSF shall perform *the key exchange* in accordance with the specified cryptographic algorithm *ECDH* and cryptographic key sizes *from 224-bit up to 512-bit* that meet the following standard: [ANS X9.63], section 5.4.1 *Standard Diffie-Hellman primitive*.

Dependencies: [FDP\_ITC.1 Import of user data without security attributes or FDP\_ITC.2 Import of user data with security attributes, or FCS\_CKM.1 Cryptographic key generation] FCS\_CKM.4 Cryptographic key destruction

Note 1: The AH3 Secure RSA/ECC/SHA library supports any valid curves over prime fields of size from 224-bit to 512-bit. However standard curves listed below whose security has been proven are in the scope of this evaluation. 1) [NIST curves]: Curves P-224, P-256, P-384, P-521 2) [Brainpool curves]: brainpoolP224r1, brainpoolP224t1, brainpoolP256r1, brainpoolP256t1, brainpoolP320r1, brainpoolP320t1, brainpoolP384r1, brainpoolP384t1, brainpoolP512r1, brainpoolP512t1, 3)[SEC-recommended curves]: secp224k1, secp224r1, secp256k1, secp256r1, secp384r1, secp521r1

Note 2: The implemented routines can be used with ephemeral or static private keys. The base point is assumed to be public.

Note 3: For full compatibility, the user is responsible to perform step 2 of [ANS X9.63], section 5.2.2.1, prior to using the ECDH\_generate function.

### 6.1.18 X25519 (DH with curve25519) (optional)

248 The X25519 library of the TOE shall meet the requirement “Cryptographic operation (FCS\_COP.1)” as specified below.

FCS\_COP.1/X25519 Cryptographic key operation

Hierarchical to: No other components.

FCS\_COP.1.1/ X25519 The TSF shall perform the *key exchange* in accordance with the specified cryptographic algorithm *X25519* and cryptographic key sizes *255bits* that meet the following standard: [RFC7748]

Dependencies: [FDP\_ITC.1 Import of user data without security attributes or FDP\_ITC.2 Import of user data with security attributes, or FCS\_CKM.1 Cryptographic key generation] FCS\_CKM.4 Cryptographic key destruction

Note 1: The implemented routines can be used with ephemeral or static private keys. The base point is assumed to be public.

### 6.1.19 Secure Hash Algorithm (SHA) (optional)

249 This SFR related to AH3 Secure RSA/ECC/SHA library for the support of RSA, ECC and SHA cryptographic operations (optional).

250 The Secure Hash Algorithm (SHA) of the TOE shall meet the requirement “Cryptographic operation (FCS\_COP.1)” as specified below.

FCS\_COP.1/SHA Cryptographic operation-SHA

Hierarchical to: No other components.

FCS\_COP.1.1/SHA The TSF shall perform *secure hash computation* in accordance with a specified cryptographic algorithm *SHA224, SHA256, SHA384 and SHA512* and cryptographic key sizes *none* that meet the following standard: [FIPS PUB 180-3].

Note 1: The AH3 Secure RSA/ECC/SHA library provides the functionalities for computation of hash values. The use of these functionalities shall not be used for keyed hash operations like HMAC or similar. If these functionalities are used for keyed hash operations like HMAC or similar, specific security improvements and DPA analysis shall be required by the operating system. The SHA224, SHA256, SHA384 and SHA512 functionalities are intended to be used only for ECDSA signature generation and verification. These functions have protection against Fault attacks but don't have protection against Side Channel Attack.

Note 2: The TOE offers the functionality of hash value computation using SHA-1, SHA-224, SHA-256, SHA-384 and SHA-512. However, only the functions related to SHA-224, SHA-256, SHA-384 and SHA-512 are in scope of this evaluation and are intended to be used only for signature generation and verification. Note that neither of the functions must be used to hash secret values. In addition, the user is responsible for the truncation or padding of the hash value as required by step e), section 7.3 and step c), section 7.4.1 of the standard cited above.

Dependencies: [FDP\_ITC.1 Import of user data without security attributes or FDP\_ITC.2 Import of user data with security attributes, or FCS\_CKM.1 Cryptographic key generation] FCS\_CKM.4 Cryptographic key destruction

### 6.1.20 Bootloader

251 The TOE Functional Requirement “Inter-TSF trusted channel (FTP\_ITC.1)” is specified as follows.

FTP_ITC.1	Inter-TSF trusted channel
Hierarchical to:	No other components.
FTP_ITC.1.1	The TSF shall provide a communication channel between itself and <i>the authorized user for using the Bootloader</i> that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.
FTP_ITC.1.2	The TSF shall permit <u>another trusted IT product</u> to initiate communication via the trusted channel.
FTP_ITC.1.3	The TSF shall initiate communication via the trusted channel for <i>deploying Loader Authentication sequence</i> .
Dependencies:	No dependencies.

252 The TOE Functional Requirement “Basic data exchange confidentiality (FDP\_UCT.1)” is specified as follows.

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

253 The TOE Functional Requirement “Data exchange integrity (FDP\_UIT.1)” is specified as follows.

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

254 The TOE Functional Requirement “Subset access control - Loader (FDP\_ACC.1/Loader)” is specified as follows.

FDP\_ACC.1/ Loader Subset access control - Loader

Hierarchical to: No other components.

FDP\_ACC.1.1/ Loader The TSF shall enforce the *Loader SFP* on

(1) *the subjects Authentication Sequence,*

(2) *the objects user data in external FLASH memory*

(3) *the operation deployment of Loader*

Dependencies: FDP\_ACF.1 Security attribute based access control.

Application Note: The TOE enforces the Loader SFP by FDP\_ITC.1, FDP\_UCT.1 and FDP\_UIT.1 and FDP\_ACF.1 to describe additional access control rules.

255 The TOE Functional Requirement “Security attribute based access control - Loader (FDP\_ACF.1/Loader)” is specified as follows.

FDP\_ACF.1/ Loader Security attribute based access control - Loader

Hierarchical to: No other components.

FDP\_ACF.1.1/ Loader The TSF shall enforce the *Loader SFP* to objects based on the following:

(1) *the subjects Bootloader with security attributes SRAM loading.*

(2) *the objects user data in external FLASH memory with security attributes SRAM loading.*

FDP\_ACF.1.2/ Loader The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *Bootloader can do loading operation in SRAM after succession of Authentication.*

FDP\_ACF.1.3/ Loader The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: *SRAM can be controlled based on security attributes ,which can be limited by Bootloader sequence.*

FDP\_ACF.1.4/ Loader The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *Bootloader can't loading the SRAM without succession of Authentication.*

Dependencies: FMT\_MSA.3 Static attribute initialisation.

### 6.1.21 Authentication Proof of Identity

256 The TOE shall meet the requirement “Authentication Proof of Identity (FIA\_API.1)” as specified below.

FIA_API.1	Authentication Proof of Identity
Hierarchical to:	No other components
Dependencies:	No dependencies.
FIA_API.1.1	The TSF shall provide an <i>authentication during sequence of Bootloader to provide the identity of the TOE to an external entity</i> to prove the identity of the TOE and the <u>initialization data (or part of data) which are used for TOE authentication verification to an external entity.</u>

### 6.1.22 Protected External Memory

257 The TOE shall meet the requirement “Data Authentication with Identity of Guarantor (FDP\_DAU.2/PM)” as specified below.

FDP_DAU.2/PM	Data Authentication with Identity of Guarantor
Hierarchical to:	FDP_DAU.1 Basic Data Authentication
Dependencies:	FIA_UID.1 Timing of identification
FDP_DAU.2.1/PM	The TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of <i>data objects and containers stored in the external memory.</i>
FDP_DAU.2.2/PM	The TSF shall provide <i>the 3S</i> with the ability to verify evidence of the validity of the indicated information and the identity of the user that generated the evidence.
Refinement:	The TSF generates the evidence that the data objects and containers stored in the external memory are generated by the dedicated 3S instance based on FDP_IRA.1/PM, FDP_SDC.1/PM and FDP_SDI.2/PM.

258 The TOE shall meet the requirement “Timing of identification (FIA\_UID.1/PM)” as specified below.

FIA_UID.1/PM	Timing of identification
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FIA_UID.1.1/PM	The TSF shall allow <i>the secure start-up or wake-up with downloading the image from the external memory</i> on behalf of the user to be performed before the user is identified.
FIA_UID.1.2/PM	The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.
Refinement:	The user is the 3S itself. The data objects and containers stored in the external memory need to be identified before any further action.

Application Note: When the image is downloaded from external passive memory the integrity of the transfer of data is checked.

259 The TOE shall meet the requirement “Replay detection (FPT\_RPL.1/PM)” as specified below.

FPT\_RPL.1/PM Replay detection

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT\_RPL.1.1/PM The TSF shall detect replay for the following entities: *commands issued by the 3S to the external memory for the read, write and erase operations*

FPT\_RPL.1.2/PM The TSF shall perform  
 1) *halt the boot procedure*  
 2) *return an error status*  
 when a replay is detected.

260 The TOE shall meet the requirement “Protection against an unauthorized rollback of memory content (FDP\_URC.1/PM)” as specified below.

FDP\_URC.1/PM Protection against an unauthorized rollback of memory content

Hierarchical to: No other components.

Dependencies: No dependencies.

FDP\_URC.1.1/PM The TOE shall detect an unauthorized replacement of the content stored in *external memory* before the content is used. The detection shall be effective in any case where modification or read operation depends on the current content of this external memory.

FDP\_URC.1.2/PM Upon detection of unauthorized rollback of the content stored in a physically separated memory, the TOE shall stop TOE operation, and return an error status

261 The TOE shall meet the requirement “Irreversibility Anchor for external memory (FDP\_IRA.1/PM)” as specified below.

FDP\_IRA.1/PM Irreversibility Anchor for external memory

Hierarchical to: No other components.

Dependencies: No dependencies.

FDP\_IRA.1.1/PM The TSF shall verify the freshness of data for each read operation from *the passive external memory*.

FDP\_IRA.1.2/PM The irreversibility anchor shall maintain a distinct transaction reference for each write, erase operation and that is unambiguously linked with the current content

of the transaction with the associated physically separated memory.

FDP\_IRA.1.3/PM The state of the irreversibility anchor implemented by the TSF shall be maintained during any operation mode.

Refinement: The passive external memory is considered outside of the TOE, even though it may be packaged together with the SoC including the 3S.

262 The TOE shall meet the requirement “Stored data confidentiality (FDP\_SDC.1/PM)” as specified below.

FDP\_SDC.1/PM Stored data confidentiality

Hierarchical to: No other components.

Dependencies: No dependencies.

FDP\_SDC.1.1/PM The TSF shall ensure the confidentiality of the information of the user data while it is stored in the *external memory*.

263 The TOE shall meet the requirement “Stored data integrity monitoring and action (FDP\_SDI.2/ PM)” as specified below.

FDP\_SDI.2/PM Stored data integrity monitoring and action

Hierarchical to: FDP\_SDI.1 Stored data integrity monitoring

Dependencies: No dependencies.

FDP\_SDI.2.1/PM The TSF shall monitor user data stored in containers controlled by the TSF for *integrity errors* on all objects, based on the following attributes: *digital signature or authentication tag*.

FDP\_SDI.2.2/tPM Upon detection of a data integrity error, the TSF shall *stop TOE operation and return an error status*.

### 6.1.23 SMK library operation

264 The TOE shall meet the requirement “Stored data confidentiality (FDP\_SDC.1/SMK)” as specified below.

FDP\_SDC.1/SMK Stored data confidentiality

Hierarchical to: No other components.

Dependencies: No dependencies.

FDP\_SDC.1.1/SMK The TSF shall ensure the confidentiality of the information of the user data while it is stored in the *OTP*.

## 6.1.24 Summary of Security Functional Requirements

Table 6-1 Security Functional Requirements defined in Smart Card IC Protection Profile

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/Test and FMT_LIM.1/Debug)
Limited availability(FMT_LIM.2/Test and FMT_LIM.2/Debug)
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)
Authentication Proof of Identity (FIA_API.1)
Inter-TSF trusted channel (FTP_ITC.1)
Basic data exchange confidentiality (FDP_UCT.1)
Data exchange integrity (FDP_UIT.1)
Subset access control - Loader (FDP_ACC.1/ Loader)
Security attribute based access control - Loader (FDP_ACF.1/Loader)
Random number generation – PTG.2(FCS_RNG.1/PTG.2)
Random number generation – DRG.3(FCS_RNG.1/DRG.3)



**Table 6-2 Augmented Security Functional Requirements**

<b>Security Functional Requirements</b>
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)
Cryptographic operation (FCS_COP.1/TDES)
Cryptographic operation (FCS_COP.1/AES)
Cryptographic operation (FCS_COP.1/KDF_KeyWrap)
Cryptographic operation (FCS_COP.1/KDF_KBKDF)
Cryptographic operation (FCS_COP.1/SHA_HW)
Cryptographic operation (FCS_COP.1/HMAC)
Cryptographic operation (FCS_COP.1/RSA) (optional)
Cryptographic key generation (FCS_CKM.1/ RSA) (optional)
Cryptographic operation (FCS_COP.1/ECDSA) (optional)
Cryptographic operation (FCS_COP.1/ECDH) (optional)
Cryptographic key generation (FCS_CKM.1/ ECDSA) (optional)
Cryptographic operation (FCS_COP.1/ X25519) (optional)
Cryptographic operation (FCS_COP.1/SHA) (optional)
Data Authentication with Identity of Guarantor (FDP_DAU.2/PM)
Timing of identification (FIA_UID.1/PM)
Replay detection (FPT_RPL.1/PM)
Protection against an unauthorized rollback of memory content (FDP_URC.1/PM)
Irreversibility Anchor for external memory (FDP_IRA.1/PM)
Stored data confidentiality (FDP_SDC.1/PM)
Stored data integrity monitoring and action (FDP_SDI.2/ PM)
Stored data confidentiality (FDP_SDC.1/SMK)

## 6.2 TOE Assurance Requirements

265 The Security Target will be evaluated according to

### Security Target evaluation (Class ASE)

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

### Evaluation Assurance Level 5 (EAL5)

267 and augmented by the following components

### ALC\_DVS.2 and AVA\_VAN.5

268 corresponding to level "EAL5+".

269 All refinements from *Protection Profile BSI-PP-0084 version 1.0* for the assurance requirements (ALC\_DEL, ALC\_DVS, ALC\_CMS, ALC\_CMC, ADV\_ARC, ADV\_FSP, ADV\_IMP, ATE\_COV, AGD\_OPE, AGD\_PRE and AVA\_VAN) have to be taken into consideration.

#### Class ADV: Development

Architectural design	(ADV_ARC.1)
Functional Specification	(ADV_FSP.5)
Implementation Representation	(ADV_IMP.1)
TSF Internals	(ADV_INT.2)
TOE Design	(ADV_TDS.4)

#### 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.5)
Delivery	(ALC_DEL.1)
<b><u>Development Security</u></b>	<b><u>(ALC_DVS.2)</u></b>
Life Cycle Definition	(ALC_LCD.1)
Tools and Techniques	(ALC_TAT.2)

#### 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.3)

Functional Tests (ATE\_FUN.1)  
Independent Testing (ATE\_IND.2)

**Class AVA: Vulnerability assessment**  
**Vulnerability Analysis** **(AVA VAN.5)**

## 6.3 Security Requirements Rationale

### 6.3.1 Rationale for the Security Functional Requirements

270 Table 6-3 below gives an overview, how the security functional requirements are combined to meet the security objectives. The detailed justification follows after the table.

**Table 6-3 Security Requirements versus Security Objectives**

Objective	TOE Security Functional and Assurance Requirements
O.Leak-Inherent	FDP_ITT.1 “Basic internal transfer protection” FPT_ITT.1 “Basic internal TSF data transfer protection” FDP_IFC.1 “Subset information flow control” AVA_VAN.5 “Advanced methodical vulnerability analysis”
O.Phys-Probing	FDP_SDC.1 “Stored data confidentiality” FPT_PHP.3 “Resistance to physical attack” FDP_SDC.1/SMK “Stored data confidentiality”
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.5 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/Test “Limited capabilities” FMT_LIM.1/Debug “Limited capabilities” FMT_LIM.2/Test “Limited availability” FMT_LIM.2/Debug “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”
O.RND	FCS_RNG.1/PTG.2 “Quality metric for random numbers” FCS_RNG.1/DRG.3 “Quality metric for random numbers” plus those for O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation, O.Leak-Forced FDP_ITT.1, FPT_ITT.1, FDP_IFC.1, FPT_PHP.3, FRU_FLT.2, FPT_FLS.1, AVA_VAN.5, ADV_ARC.1

Objective	TOE Security Functional and Assurance Requirements
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"
O.TDES	FCS_COP.1/TDES
O.AES	FCS_COP.1/ AES
O.KDF	FCS_COP.1/KDF_KeyWrap FCS_COP.1/KDF_KBKDF
O.RSA	FCS_COP.1/RSA FCS_CKM.1/RSA
O.ECDSA	FCS_COP.1/ ECDSA FCS_CKM.1/ ECDSA
O.ECDH	FCS_COP.1/ ECDH
O.X25519	FCS_COP.1/X25519
O.SHA	FCS_COP.1/SHA (SHA2) FCS_COP.1/SHA_HW (SHA2/3)
O.HMAC	FCS_COP.1/HMAC
O.Authentication	FIA_API.1 " Authentication Proof of Identity"
OE.TOE_Auth	not applicable
O.Ctrl_Auth_Loader	FTP_ITC.1 "Inter-TSF trusted channel" FDP_UCT.1 "Basic data exchange confidentiality" FDP_UIT.1 "Data exchange integrity" FDP_ACC.1/Loader "Subset access control - Loader" FDP_ACF.1/Loader "Security attribute based access control - Loader"
OE.Loader_Usage	not applicable
O.External-Content-Protection	FDP_SDC.1/PM for confidentiality protection FDP_SDI.2/PM for integrity protection
O.Mem-Command-Replay-Prot	FPT_RPL.1/PM for Replay detection
O.Mem-Irreversibility-Anchor	FDP_IRA.1/PM for Irreversibility Anchor of external memory content
O.Mem-Unauthorized-Rollback-Prot	FDP_URC.1/PM for Protection against an unauthorized rollback of content

Objective	TOE Security Functional and Assurance Requirements
O.Mem-Clone-Replace-Protection	FDP_DAU.2/PM for Data Authentication with Identity of Guarantor FIA_UID.1/PM for Timing of identification

- 271 The justification related to the security objective “Protection against Inherent Information Leakage (O.Leak-Inherent)” is as follows:
- 272 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.
- 273 It is possible that the TOE needs additional support by the Security IC Embedded Software (e.g. timing attacks are possible if the processing time of algorithms implemented in the software depends on the content of secret). This support must be addressed in the Guidance Documentation. Together with this FPT\_ITT.1, FDP\_ITT.1 and FDP\_IFC.1 are suitable to meet the objective.
- 274 The justification related to the security objective “Protection against Physical Probing (O.Phys-Probing)” is as follows:
- 275 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.
- 276 It is possible that the TOE needs additional support by the Security IC Embedded Software (e. g. to send data over certain buses only with appropriate precautions). This support must be addressed in the Guidance Documentation. Together with this FPT\_PHP.3 is suitable to meet the objective.
- 277 The justification related to the security objective “Protection against Malfunctions (O.Malfunction)” is as follows:
- 278 The definition of this objective shows that it covers a situation, where malfunction of the TOE might be caused by the operating conditions of the TOE (while direct manipulation of the TOE is covered O.Phys-Manipulation). There are two possibilities in this situation: Either the operating conditions are inside the tolerated range or at least one of them is outside of this range. The second case is covered by FPT\_FLS.1, because it states that a secure state is preserved in this case. The first case is covered by FRU\_FLT.2 because it states that the TOE operates correctly under normal (tolerated) conditions. The functions implementing FRU\_FLT.2 and FPT\_FLS.1 must work independently so that their operation cannot be affected by the Security IC Embedded Software (refer to the refinement). Therefore, there is no possible instance of conditions under O.Malfunction, which is not covered.
- 279 The justification related to the security objective “Protection against Physical Manipulation (O.Phys-Manipulation)” is as follows:
- 280 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.
- 281 It is possible that the TOE needs additional support by the Embedded Software (for instance by implementing FDP\_SDI.1 to check data integrity with the help of appropriate checksums, refer to Section

6.1). This support must be addressed in the Guidance Documentation. Together with this FPT\_PHP.3 is suitable to meet the objective.

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

283 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.

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

285 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/Test and FMT\_LIM.2/Debug, and the second one by FMT\_LIM.1/Test and FMT\_LIM.1/Debug. 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.

286 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 6-3.

287 It was chosen to define FMT\_LIM.1/Test, FMT\_LIM.1/Debug, FMT\_LIM.2/Test and FMT\_LIM.2/Debug 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.

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

289 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.

290 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.

291 The objective must be supported by organisational and other measures, which the TOE Manufacturer has to implement. These measures are a subset of those measures, which are examined during the evaluation

---

of the assurance requirements of the classes AGD and ALC.

- 292 The justification related to the security objective “Random Numbers (O.RND)” is as follows:
- 293 FCS\_RNG.1 requires the TOE to provide random numbers of good quality. The metrics associated to the DTRNG are given by the SFRs FCS\_RNG.1/PTG.2 and DRG.3.
- 294 Other security functional requirements, which prevent physical manipulation and malfunction of the TOE (see the corresponding objectives listed in the table), support this objective because they prevent attackers from manipulating or otherwise affecting the random number generator.
- 295 Random numbers are often used by the Security IC Embedded Software to generate cryptographic keys for internal use. Therefore, the TOE must prevent the unauthorised disclosure of random numbers. Other security functional requirements which prevent inherent leakage attacks, probing and forced leakage attacks ensure the confidentiality of the random numbers provided by the TOE.
- 296 Depending on the functionality of specific TOEs the Security IC Embedded Software will have to support the objective by providing runtime-tests of the random number generator. Together, these requirements allow the TOE to provide cryptographically good random numbers and to ensure that no information about the produced random numbers is available to an attacker.
- 297 It was chosen to define FCS\_RNG.1 explicitly, because Part 2 of the Common Criteria does not contain generic security functional requirements for Random Number generation. (Note, that there are security functional requirements in Part 2 of the Common Criteria, which refer to random numbers. However, they define requirements only for the authentication context, which is only one of the possible applications of random numbers.)
- 298 The security objective Access control and authenticity for the Loader (O.Ctrl\_Auth\_Loader) is covered by the SFR as follows:
- 299 The SFR FDP\_ACC.1/Loader defines the subjects, objects and operations of the Loader SFP enforced by the SFR FTP\_ITC.1, FDP\_UCT.1, FDP\_UIT.1, FDP\_ACF.1/Loader
- 300 The SFR FTP\_ITC.1 requires the TSF to establish a trusted channel with assured identification of its end points and protection of the channel data from modification or disclosure.
- 301 The SFR FDP\_UCT.1 requires the TSF to receive data protected from unauthorised disclosure.
- 302 The SFR FDP\_UIT.1 requires the TSF to verify the integrity of the received user data.
- 303 The SFR FDP\_ACF.1/Loader requires the TSF to implement access control for the Loader functionality.
- 304 The FCS\_COP.1/TDES meets the security objective “Cryptographic service Triple-DES (O.TDES)”.
- 305 The FCS\_COP.1/AES meets the security objective “Cryptographic service AES (O.AES)”.
- 306 The FCS\_COP.1/KDF\_KeyWrap and FCS\_COP.1/KDF\_KBKDF meets the security objective “Cryptographic service Key Derivation Function (O.KDF)”.
- 307 The FCS\_COP.1/SHA\_HW meets the security objective “Cryptographic service SHA2 and SHA3 (O.SHA)”.
- 308 The FCS\_COP.1/HMAC meets the security objective “Cryptographic service Keyed-Hash Message Authentication Code (O.HMAC)”.
- 309 The security functional requirement(s) “Cryptographic operation (FCS\_COP.1/RSA,FCS\_COP.1/ECDSA,ECDH,X25519)” exactly requires those functions to be
-



implemented which are demanded by O.RSA or O.ECDSA, O.ECDH, O.X25519, FCS\_CKM.1 supports the generation of keys needed for this cryptographic operations(optional). Therefore, FCS\_COP.1/RSA, FCS\_COP.1/ECDSA, FCS\_COP.1/ECDH, FCS\_COP.1/X25519, FCS\_CKM.1/RSA and FCS\_CKM.1/ECDSA are suitable to meet the security objective.

- 310 The FCS\_COP.1/SHA meet the security objective “Cryptographic service SHA2 (O.SHA)”.
- 311 The security objective “Authentication to external entities (O.Authentication) is directly covered by the SFR FIA\_API.1.
- 312 The justification related to the security objective “Area based Memory Access Control (O.Mem-Access)” is as follows:
- 313 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.
- 314 The security functional requirement “Security attribute based access control (FDP\_ACF.1)” with the related Security Function Policy (SFP) “Memory Access Control Policy” exactly requires the implementation of an area based memory access control, which is a requirement from O.Mem-Access. Therefore, FDP\_ACF.1 with its SFP is suitable to meet the security objective.
- 315 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.
- 316 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.
- 317 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.
- 318 The justification related to the security objective “Protection during Packaging, Finishing and Personalisation (OE.Process-Sec-IC)” is as follows:
- 319 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.
- 320 The justification related to the security objective “Protection against unauthorized disclosure and undetected modification of external memory content (O.External-Content-Protection)” is as follows:
- 321 The SFR FDP\_SDC.1/PM ensures protection of confidentiality of the content stored in the external memory, while the SFR FDP\_SDI.2/PM ensures protection of the integrity of the content stored in the external memory. Since the protection is under full control inside the 3S also the transfer between the 3S and the external memory is protected. Therefore, these security functional requirements support the objective.
- 322 The justification related to the security objective “Protection against replay of commands between the 3S and the external memory (O.Mem-Command-Replay-Prot)” is as follows:
-

- 323 The SFR FPT\_RPL.1/PM requires the TSF to detect replayed transactions (read, write and erase operations) to the external memory. This requirement is considered in the assignment of FPT\_RPL.1.1/PM. Therefore, this security functional requirement supports the objective. The action on a detected transaction replay is left to the ST author since it depends on the application context.
- 324 The justification related to the security objective “Protection against content (O.Mem-Unauthorized-Rollback-Prot)” is as follows:
- 325 The SFR FDP\_URC.1/PM requires that the TSF detects the case when the content of the external memory has been replaced by previous versions of them. This way, this security functional requirement supports the objective.
- 326 The justification related to the security objective “External memory content Irreversibility Anchor (O.Mem-Irreversibility-Anchor)” is as follows:
- 327 The SFR FDP\_IRA.1/PM requires the TOE to implement distinct transaction references for each write and erase operation that is unambiguously linked with the current content of the transaction with the external memory. Thereby, the data freshness can be verified during a read operation based on the data maintained by the irreversible anchor. If the external memory is a non-volatile memory, the Irreversibility-Anchor needs to be maintained in any operation mode. By providing the mechanism required by this SFR, the security objective O.Mem-Irreversibility-Anchor is directly supported.
- 328 The justification related to the security objective “Protection against external memory cloning or replacement (O.Mem-Clone-Replace-Protection)” is as follows:
- 329 The SFR FDP\_DAU.2/PM requires the TOE to be able to generate evidence that guarantees the validity of data objects and containers stored in the external memory. With the refinement that the dedicated 3S instance is the user in case of user data the cloning or replacement of the external memory is detected. The SFR FIA\_UID.1/PM requires the definition of actions that can be performed without user identification. The authenticity external memory content needs to be identified instead of a user. This is described in a refinement for this SFR. The authenticity of the data stored in the external memory needs to be identified before any user data is accessed. By providing the mechanism required by these two SFRs, the security objective O.Mem-Clone-Replace-Protection is directly supported.
- 330 The SFR FDP\_SDC.1/SMK ensures protection of confidentiality of the Application’s encryption key stored in the OTP.

### 6.3.2 Dependencies of Security Functional Requirements

- 331 Table 6-4 below lists the security functional requirements defined in this Security Target, their dependencies and whether they are satisfied by other security requirements defined in this Security Target. The text following the table discusses the remaining cases.

**Table 6-4 Dependencies of the Security Functional Requirements**

Security Functional Requirement	Dependencies	Fulfilled by security requirements
FRU_FLT.2	FPT_FLS.1	Yes
FPT_FLS.1	None	No dependency
FMT_LIM.1/Test, FMT_LIM.1/Debug	FMT_LIM.2/Test, FMT_LIM.2/Debug	Yes

Security Functional Requirement	Dependencies	Fulfilled by security requirements
FMT_LIM.2/Test, FMT_LIM.2/Debug	FMT_LIM.1/Test, FMT_LIM.1/Debug	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_IFT.1	See discussion below
FPT_ITT.1	None	No dependency
FCS_RNG.1/PTG.2	None	No dependency
FCS_RNG.1/DRG.3	None	No dependency
FCS_COP.1 /TDES	FCS_CKM.4	Yes (by environment, see discussion below)
	FDP_ITC.1 or FDP_ITC.2 (if not FCS_CKM.1) or FCS_CKM.1	Yes (by environment, see discussion below)
FCS_COP.1 /AES	FCS_CKM.4	Yes (by environment, see discussion below)
	FDP_ITC.1 or FDP_ITC.2 (if not FCS_CKM.1) or FCS_CKM.1	Yes (by environment, see discussion below)
FCS_COP.1 /KDF	FCS_CKM.4	Yes (by environment, see discussion below)
	FDP_ITC.1 or FDP_ITC.2 (if not FCS_CKM.1) or FCS_CKM.1	Yes (by environment, see discussion below)
FCS_COP.1 /SHA_HW	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1, FCS_CKM.4	See discussion below
FCS_COP.1 /HMAC	FCS_CKM.4	Yes (by environment, see discussion below)
	FDP_ITC.1 or FDP_ITC.2 (if not FCS_CKM.1) or FCS_CKM.1	Yes (by environment, see discussion below)
FCS_CKM.1 /RSA (optional)	FCS_COP.1 or FCS_CKM.2	Yes
	FCS_CKM.4	Yes (by environment, see discussion below)
FCS_COP.1/RSA (optional)	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	Yes
	FCS_CKM.4	Yes (by environment, see discussion below)

Security Functional Requirement	Dependencies	Fulfilled by security requirements
FCS_COP.1/ECDSA (optional)	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	Yes
	FCS_CKM.4	Yes (by environment, see discussion below)
FCS_COP.1/ECDH (optional)	FDP_ITC.1 or FDP_ITC.2, or FCS_CKM.1	Yes
	FCS_CKM.4	Yes (by environment, see discussion below)
FCS_COP.1/X25519 (optional)	FCS_CKM.4	Yes (by environment, see discussion below)
	FDP_ITC.1 or FDP_ITC.2 (if not FCS_CKM.1) or FCS_CKM.1	Yes (by environment, see discussion below)
FCS_CKM.1 /ECDSA (optional)	FCS_COP.1 or FCS_CKM.2	Yes
	FCS_CKM.4	Yes (by environment, see discussion below)
FCS_COP.1/SHA (optional) FCS_COP.1/ECDSA (optional)	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1,FCS_CKM.4	See discussion below
	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	Yes
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
FTP_ITC.1	None	No dependency
FDP_UCT.1	FTP_ITC.1 or FTP_TRP.1, FDP_ACC.1 or FDP_IFC.1	Yes
FDP_UIT.1	FTP_ITC.1 or FTP_TRP.1, FDP_ACC.1 or FDP_IFC.1	Yes
FDP_ACC.1/ Loader	FDP_ACF.1	Yes
FDP_ACF.1/ Loader	FMT_MSA.3	See discussion below
FIA_API.1	None	No dependency
FDP_SDC.1/PM	None	No dependency
FDP_UID.1/PM	None	No dependency

Security Functional Requirement	Dependencies	Fulfilled by security requirements
FDP_SDI.2/PM	None	No dependency
FPT_RPL.1/PM	None	No dependency
FDP_IRA.1/PM	None	No dependency
FDP_URC.1/PM	None	No dependency
FDP_DAU.2/PM	FIA_UID.1	Satisfied by FIA_UID.1
FDP_SDC.1/SMK	None	No dependency

- 332 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.
- 333 In particular the security functional requirements providing resistance of the hardware against manipulations (e. g. FPT\_PHP.3) support all other more specific security functional requirements (e. g. FCS\_RNG.1) because they prevent an attacker from disabling or circumventing the latter. Together with the discussion of the dependencies above this shows that the security functional requirements build a mutually supportive whole.
- 334 The functional requirements FCS\_CKM.1 and FCS\_CKM.4 which are dependent to FCS\_COP.1/TDES and FCS\_COP.1/AES are not included in this Security Target since the TOE only provides an engine for encryption and decryption. But the Security IC Embedded Software may fulfill these requirements related to the needs of the implemented application. The dependent requirements of FCS\_COP.1/TDES and FCS\_COP.1/AES concerning these functions shall be fulfilled by the environment (Security IC Embedded Software).
- 335 The functional requirements FCS\_CKM.1 and FCS\_CKM.4 which are dependent to FCS\_COP.1/HMAC is not included in this Security Target since the TOE only provides an engine for message digesting. But the Security IC Embedded Software may fulfill these requirements related to the needs of the implemented application. The dependent requirements of FCS\_COP.1/HMAC concerning these functions shall be fulfilled by the environment (Security IC Embedded Software).
- 336 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.
- 337 The dependency FMT\_MSA.3 of FDP\_ACF.1/Loader is not be necessary. The security attributes of ROM used to enforce the Loader SFP are fixed by the IC manufacturer. The access attribute of ROM have DEFAULT value.
- 338 The TOE provides the cryptographic key generation for RSA and ECDSA by the TOE (FCS\_CKM.1/RSA, FCS\_CKM.1/ECDSA), but it is up to the Security IC Embedded Software's security policy to adopt the cryptographic key generation by the TOE or use the cryptographic key generation by the Secure Sub-System Embedded Software. The dependent requirements of FCS\_COP.1/RSA and FCS\_COP.1/ECDSA shall be fulfilled by the environment (Security IC Embedded Software).

- 339 The functional requirement FCS\_CKM.1 which is dependent to FCS\_COP.1/ECDH and FCS\_COP.1/X25519 are not included in this Security Target. But the Security IC Embedded Software may fulfil this requirement related to the needs of the implemented application. The dependent requirements of FCS\_COP.1/ECDH and FCS\_COP.1/X25519 concerning this function shall be fulfilled by the environment (Security IC Embedded Software).
- 340 The functional requirement FCS\_CKM.4 which is dependent to FCS\_COP.1/RSA, FCS\_COP.1/ECDH, FCS\_COP.1/ECDSA and FCS\_COP.1/X25519 are not included in this Security Target. But the Security IC Embedded Software may fulfil this requirement related to the needs of the implemented application. The dependent requirements of FCS\_COP.1/RSA, FCS\_COP.1/ECDH, FCS\_COP.1/ECDSA and FCS\_COP.1/X25519 concerning this function shall be fulfilled by the environment (Security IC Embedded Software).
- 341 Since SHA is a keyless algorithm, there is no need for key import as required by dependency to FDP\_ITC.1, FDP\_ITC.2 or key generation as required by dependency to FCS\_CKM.1 or destruction as required by dependency to FCS\_CKM.4. So the dependencies to FDP\_ITC.1, FDP\_ITC.2, FCS\_CKM.1 and FMT\_CKM.4 are not required.
- 342 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. All data requires the same protection and is only transferred between two instances as stated in the Data Processing Policy. Therefore, 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).

### 6.3.3 Rationale for the Assurance Requirements

- 343 The assurance level EAL5 and the augmentation with the requirements ALC\_DVS.2, and AVA\_VAN.5 were chosen in order to meet assurance expectations explained in the following paragraphs.
- 344 An assurance level of EAL5 is required for this type of TOE since it is intended to defend against sophisticated attacks. This evaluation assurance level was selected since it is designed to permit a developer to gain maximum assurance from positive security engineering based on good commercial practices. In order to provide a meaningful level of assurance that the TOE provides an adequate level of defence against such attacks, the evaluators should have access to the low level design and source code.

#### ALC\_DVS.2 Sufficiency of Security Measures

- 345 Development security is concerned with physical, procedural, personnel and other technical measures that may be used in the development environment to protect the TOE.
- 346 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.
- 347 This assurance component is a higher hierarchical component to EAL5 (which only requires ALC\_DVS.1). ALC\_DVS.2 has no dependencies.

**AVA\_VAN.5 Advanced Methodical Vulnerability Analysis**

- 348 Due to the intended use of the TOE, it must be shown to be highly resistant to penetration attacks. This assurance requirement is achieved by the AVA\_VAN.5 component.
- 349 Independent vulnerability analysis is based on highly detailed technical information. The main intent of the evaluator analysis is to determine that the TOE is resistant to penetration attacks performed by an attacker possessing high attack potential.
- 350 AVA\_VAN.5 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: Basic design”.
- 351 All these dependencies are satisfied by EAL5.
- 352 It has to be assumed that attackers with high attack potential try to attack Security ICs like smart cards used for digital signature applications or payment systems. Therefore, specifically AVA\_VAN.5 was chosen in order to assure that even these attackers cannot successfully attack the TOE

**6.3.4 Security Requirements are Internally Consistent**

- 353 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.
- 354 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.
- 355 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/Test, FMT\_LIM.2/Debug, FCS\_RNG.1 and those implemented in the Security IC Embedded Software.
- 356 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/Test, FMT\_LIM.1/Debug, FMT\_LIM.2/Test, FMT\_LIM.2/Debug, FCS\_RNG.1 and those implemented in the Security IC Embedded Software.
- 357 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).

- 358 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/Test and FMT\_LIM.2/Debug) 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.
- 359 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/Test and FMT\_LIM.2/Debug) 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.
- 360 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/Test, FMT\_LIM.1/Debug, FMT\_LIM.2/Test and FMT\_LIM.2/Debug 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.
- 361 The combination of the security functional requirements FMT\_LIM.1/Test, FMT\_LIM.1/Debug, FMT\_LIM.2/Test and FMT\_LIM.2/Debug ensures that (especially after TOE Delivery) these additional functions cannot be abused by an attacker to (i) disclose or manipulate user data of the Composite TOE, (ii) to manipulate (explore, bypass, deactivate or change) security features or services of the TOE or of the Security IC Embedded Software or (iii) to enable other attacks on the assets. Hereby the binding between these two security functional requirements is very important:
- 362 The security functional requirement Limited Capabilities (FMT\_LIM.1/Test and FMT\_LIM.1/Debug) must close gaps which could be left by the control being applied to the function's interface (Limited Availability (FMT\_LIM.2/Test and FMT\_LIM.2/Debug)). Note that the security feature or services which limits the availability can be bypassed, deactivated or changed by physical manipulation or a malfunction caused by an attacker. Therefore, if Limited Availability (FMT\_LIM.2/Test and FMT\_LIM.2/Debug) is vulnerable, it is important to limit the capabilities of the functions in order to limit the possible benefit for an attacker.
- 363 The security functional requirement Limited Availability (FMT\_LIM.2/Test and FMT\_LIM.2/Debug) must close gaps which could result from the fact that the function's kernel in principle would allow to perform attacks. The TOE must limit the availability of functions which potentially provide the capability to disclose or manipulate user data of the Composite TOE, to manipulate security features or services of the TOE or of the Security IC Embedded Software or to enable other attacks on the assets. Therefore, if an attacker could benefit from using such functions, it is important to limit their availability so that an attacker is not able to use them.
- 364 No perfect solution to limit the capabilities (FMT\_LIM.1/Test and FMT\_LIM.1/Debug) is required if the limited availability (FMT\_LIM.2/Test and FMT\_LIM.2/Debug) alone can prevent the abuse of functions. No perfect solution to limit the availability (FMT\_LIM.2/Test and FMT\_LIM.2/Debug) is required if the limited
-



capabilities (FMT\_LIM.1/Test and FMT\_LIM.1/Debug) 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.

- 365 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.
- 366 The security functional requirements required to meet the security objectives O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation and O.Leak-Forced also protect the cryptographic algorithms implemented according to the security functional requirement FCS\_COP.1. Therefore, these security functional requirements support the secure implementation and operation of FCS\_COP.1.
- 367 Parts of the Security 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.
- 368 Two refinements from the PP [5] have to be discussed here in the ST as the assurance level is increased. The refinement for ALC\_CMS from the PP [5] can even be applied at the assurance level EAL5 augmented with ALC\_CMS.5. The assurance component ALC\_CMS.4 is augmented to ALC\_CMS.5 with aspects regarding the configuration control system for the TOE. The refinement is not touched. The refinement for ADV\_FSP from the PP [5] can even be applied at the assurance level EAL5 augmented with ADV\_FSP.5. The assurance component ADV\_FSP.4 is extended to ADV\_FSP.5 with aspects regarding the description level. The level is increased from informal to semi-formal with informal description. The refinement is not touched by this measure.

# 7 TOE SUMMARY SPECIFICATION

369 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**

- 370 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-defined appropriate actions by software the TOE.
- 371 The secure state is maintained by TOE's detectors. The TOE's detectors are monitoring the failure occurs. If the failures are happening, the TOE goes into IRQ state. This satisfies the FPT\_FLS.1 "Failure with preservation of secure state."

#### TOE's Detectors

- 372 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 generates immediately interrupt when an event is detected.
  - Or, a special function register bit is set.

TOE's detectors are implemented by the hardware. The detection cannot be affected or bypassed by Security IC Embedded Software. The reaction to the detection can be configured by the software.

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

### **SFR2: FRU\_FLT.2: Limited fault tolerance**

- 374 These Integrity Checkers are used for preventing noise and laser from causing undefined or unpredictable behaviour of the chip.
- ECC/ Parity /CRC32 Calculator

### **SFR3: FPT\_PHP.3: Resistance to physical attacks**

- 375 This requirement is achieved by security feature as the shield must be removed and bypassed in order to perform physical intrusive attacks.

### **SFR4: FDP\_ACC.1: Subset access control**

- 376 This requirement is achieved by security register access control, invalid address access and access right for the code executed in the internal SRAM.

### **SFR5: FDP\_ACF.1: Security attributes based access control.**

- 377 This is covered by the Privilege and User modes of the TOE.

**SFR6: FMT\_MSA.3: Static attribute initialization.**

378 All Special Function Registers including MPU have DEFAULT values after Power on Reset.

The access attribute of ROM and SRAM memory have DEFAULT values: ReadOnly attribute for ROM and ReadWrite attribute for SRAM.

**SFR7: FMT\_MSA.1: Management of security attributes.**

379 This is achieved with the MPU feature.

The Memory Protection Unit (MPU) enables user to partition memory and set individual protection attributes for each partition.

**SFR8: FMT\_SMF.1: Specification of management functions.**

380 This is achieved via access to Special Function Registers of Memory Protection Unit (MPU).

**SFR9: FAU\_SAS.1: Audit Storage**

381 This is fulfilled by the traceability/identification data written once and for all during the manufacturing process.

**SFR10: FMT\_LIM.1: Limited capabilities****FMT\_LIM.1/Test**

382 TEST mode can be accessed only by the TEST administrator through a proprietary protocol.

**FMT\_LIM.1/Debug**

383 Debug mode can be accessed only by the Debugger in Debugging step.

**SFR11: FMT\_LIM.2: Limited availabilities****FMT\_LIM.2/Test**

384 TEST mode can be accessed only by the TEST administrator through a proprietary protocol.

**FMT\_LIM.2/Debug**

385 Debug mode can be accessed only by the Debugger in Debugging step.

**SFR12: FDP\_IFC.1: Subset information flow control**

386 Memory Encryption: This 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.

Shield: This requirement is achieved by security feature as the shield must be removed and bypassed in order to perform physical intrusive attacks.

**SFR13: FDP\_ITT.1: Basic internal transfer protection**

387 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
- 2) Dynamic Data encryption for bus
- 3) Memory encryption
- 4) Synthesizable processor core
- 5) De-synchronization and signal-to-noise ratio reduction mechanisms

**SFR14: FPT\_ITT.1: Basic internal TSF data transfer protection**

388 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
- 2) Dynamic Data encryption for bus
- 3) Memory encryption
- 4) Synthesizable processor core
- 5) De-synchronization and signal-to-noise ratio reduction mechanisms

**SFR15: FCS\_RNG.1: Random number generation**

FCS\_RNG.1/PTG.2

389 This requirement is ensured by the design of the random number generation algorithm that makes use of Digital True Random Number Generator (DTRNG) and the associated DTRNG library conforming to *BSI-AIS-20/31 Class PTG.2* requirements (German scheme).

FCS\_RNG.1/DRG.3

390 This requirement is ensured by the design of the deterministic random bit generation algorithm that makes use of Deterministic Random Bit Generator (DRBG) library conforming to *BSI-AIS-20/31 Class DRG.3* requirements (German scheme).

**SFR16: FCS\_COP.1: Cryptographic operation**

391 This requirement is covered by the TOE.

**Triple Data Encryption Standard Engine**

392 This function is used for encrypting and decrypting data using the Triple DES symmetric algorithm with 112 bit or 168 bit key size. (FCS\_COP.1/TDES)

**AES (Advanced Encryption Standard)**

393 This function supports the AES operation with 128 bit, 192 bit and 256 bit key size. (FCS\_COP.1/AES)

**KDF (Key Derivation Function, Key Manager)**

394 This function supports the Key derivation function of AES operation (FCS\_COP.1/KeyWrap) and HMAC operation. (FCS\_COP.1/KBKDF)

**SHA2/3 (Secure Hash Algorithm)**

395 This function supports to calculate hash (digest) values. (FCS\_COP.1/SHA\_HW)

**HMAC (Keyed-Hash Message Authentication Code)**

396 This function supports to calculate keyed-hash (digest) values. (FCS\_COP.1/HMAC)

397 TORNADO-H RSA Cryptographic Library as part of AH3 Secure RSA/ECC/SHA library (optional)

398 This function assists in the acceleration of modulo exponentiations required in the RSA encryption/decryption algorithm. (FCS\_COP.1/RSA)

399 TORNADO-H is a high speed modular multiplication coprocessor for the support of the RSA public key cryptosystem. The AH3 Secure RSA/ECC/SHA library is the software built on the TORNADO-H coprocessor that provides high level interface for RSA-based algorithms.

400 TORNADO-H ECC Cryptographic Library as part of AH3 Secure RSA/ECC/SHA library (optional)

401 This function assists in the acceleration of required for the ECC cryptographic operations including the ECDSA signature generation/verification, the ECDH secret key derivation and the X25519 secret key derivation. (FCS\_COP.1/ECDSA, FCS\_COP.1/ECDH and FCS\_COP.1/X25519)

402 AH3 Secure RSA/ECC/SHA library provides a set of functions to implement elliptic curve cryptographic algorithms. In particular, it provides some functions to implement the ECDSA signature generation/verification, the ECDH secret key derivation and the X25519 secret key derivation.

403 The AH3 Secure RSA/ECC/SHA library provides the functions to calculate hash (digest) values using the SHA1, SHA224, SHA256, SHA384 and SHA 512 algorithm as specified in [FIPS PUB 180-3]

**SFR17: FCS\_CKM.1: Cryptographic key generation**

404 This requirement is covered by the TOE for the RSA/ECC key generation. (optional)

**SFR18: Reserved**

**SFR19: Reserved**

**SFR20: Inter-TSF trusted channel (FTP\_ITC.1)**

405 This requirement is achieved by processing the Authentication sequence.

**SFR21: Basic data exchange confidentiality (FDP\_UCT.1)**

406 This requirement is achieved by secure external FLASH loading.

**SFR22: Data exchange integrity (FDP\_UIT.1)**

407 This requirement is achieved by checking the checksum.

**SFR23: Subset access control - Loader (FDP\_ACC.1/ Loader)**

408 This requirement is achieved by following functions.

409 Access attribute control of Bootloader

410 SRAM memory attribute as "Disable code execution on RAM"

**SFR24: Security attribute based access control - Loader (FDP\_ACF.1/Loader)**

This is covered by the ROM Booting and SRAM Booting mode of the TOE.

**SFR25: Stored data confidentiality (FDP\_SDC.1)**

411 This requirement is achieved by the combination of the TOE security features TOE features 1) to 9) as it is unpractical to get access to internal signals and interpret them.

- 1) Static Address/Data scrambling for bus and memory
- 2) Dynamic Data encryption for bus
- 3) Memory encryption
- 4) Invalid address access
- 5) Active shield
- 6) Life time detector
- 7) Non-reversibility of TEST, Debug and NORMAL modes
- 8) Control of Booting mode

**SFR26: Stored data integrity monitoring and action (FDP\_SDI.2)**

412 This requirement is achieved by following functions.

413 RAM ECC

414 TRAM parity

**SFR27: Authentication Proof of Identity (FIA\_API.1)**

415 This requirement is achieved by processing the Authentication sequence.

**SFR28: Data Authentication with Identity of Guarantor (FDP\_DAU.2/PM)**

416 The image of the FW is stored with a certificate and the TOE stores a means to verify the public key.

417 For the data stored in the DRAM during operation of the TOE, the authenticity is determined by the AES.

**SFR29: Timing of identification (FIA\_UID.1/PM)**

- 418 The authenticity is checked by verification of the public key and digital signature that are stored with the FW.
- 419 For the data stored in DRAM during operation of the TOE, it is downloaded immediately after the return from low power mode.

**SFR30: Replay detection (FPT\_RPL.1/PM)**

- 420 After downloading the image from external Passive NVM the integrity is verified with a SHA512, and next the signature is verified.
- 421 When during operation time the image containing the firmware is downloaded from the DRAM, replay is detected when the AES.

**SFR31: Protection against an unauthorized rollback of memory content (FDP\_URC.1/PM)**

- 422 A set of rollback counters implemented.

**SFR32: Irreversibility Anchor for external memory (FDP\_IRA.1/PM)**

- 423 The TOE implements a number of rollback counters.

**SFR33: Stored data confidentiality (FDP\_SDC.1/PM)**

- 424 The image that can be downloaded from the external passive NVM is stored encrypted.
- 425 During operation of the TOE the firmware can be temporarily uploaded to and downloaded from DRAM.

**SFR34: Stored data integrity monitoring and action (FDP\_SDI.2/PM)**

- 426 The integrity of the firmware stored in external passive memory is verified.

**SFR35: Stored data confidentiality (FDP\_SDC.1/SMK)**

- 427 The encryption key for the application image is encrypted and stored in the OTP.



# 8 Annex

## 8.1 Glossary

### Application Data

- 428 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

- 429 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

- 430 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

- 431 User of the Composite Product in Phase 7.

### IC Dedicated Software

- 432 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

- 433 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

- 434 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

- 435 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)**

- 436 Electronic component(s) designed to perform processing and/or memory functions.

**Pre-personalisation Data**

- 437 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**

- 438 Composition of the TOE, the Security IC Embedded Software, User Data and the package (the Security IC carrier).

**Security IC Embedded Software**

- 439 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**

- 440 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**

- 441 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**

- 442 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**

- 443 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**

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

## 8.2 Abbreviations

Table 8-1 Abbreviations

Abbreviations	Descriptions
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 Feature
TSFI	TSF Interface
TSP	TOE Security Policy
3S	Secure Sub-System

### 8.3 References

- [1] Common Criteria, Part 1: Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model, Version 3.1, Revision 5, April 2017, CCMB-2017-04-001
- [2] Common Criteria, Part 2: Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components, Version 3.1, Revision 5, April 2017, CCMB-2017-04-001
- [3] Common Criteria, Part 3: Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Components, Version 3.1, Revision 5, April 2017, CCMB-2017-04-001
- [4] Common Methodology for Information Technology Security Evaluation, Evaluation Methodology, Version 3.1, Revision 5, April 2017, CCMB-2017-04-001
- [5] Eurosmart Security IC Platform Protection Profile with Augmentation Packages, Version 1.0, BSI-CC-PP-0084-2014.
- [6] AIS-20/31: Functionality classes and evaluation methodology for true (physical) random number generators, Version 1, 25.09.2001, Bundesamt für Sicherheit in der Informationstechnik
- [7] A proposal for: Functionality classes for random number generators, Version 2.0, 18.09.2011, Bundesamt für Sicherheit in der Informationstechnik
- [8] ALGO: Federal Gazette No 19, Notification in accordance with the Electronic Signatures Act and the Electronic Signatures Ordinance (overview of suitable algorithms), Federal Network Agency for Electricity, Gas, Telecommunications, Post and Railway, 2008-11-17
- [9] [FIPS SP800-67] Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher, Revision 2
- [10] [FIPS 197] Advanced Encryption Standard (AES), 2001-11-26
- [11] [ISO/IEC 14888-2:2008] - Information technology -- Security techniques-- Digital signatures with appendix -- Part 2: Integer factorization based mechanisms.
- [12] CC Supporting Document, Mandatory Technical Document, "Application of Attack Potential to Smartcards": version 2.9 (January 2013) as recommended by SOG-IS.
- [13] [FIPS PUB 180-3] U.S. Department of Commerce / National Bureau of Standards, Secure Hash Algorithm, FIPS PUB 180-3, 2008-October
- [14] [NIST curves] Federal Information Processing Standards Publication FIPS PUB 180-3, Digital Signature Standard; U.S. department of Commerce / National Institute of Standards and Technology (NIST), June 2009
- [15] [ETSI TS 102 176-1] Electronic Signatures and Infrastructures (ESI); Algorithms and Parameters for Secure Electronic Signatures; Part 1: Hash functions and asymmetric algorithms, 2007-11, version 2.0.0
- [16] [SCA on Prime Gen] T. Finke, M. Gebhardt and W. Schindler, A New Side-Channel Attack on RSA Prime Generation, CHES 2009, LNCS 5747, pp. 141-155, 2009.
- [17] Les règles et recommandations concernant le choix et le dimensionnement de mécanismes cryptographiques. Annexe B1 du RGS 2.0. Version 2.03, 21/02/2014, ANSSI.  
[http://www.ssi.gouv.fr/uploads/2015/01/RGS\\_v-2-0\\_B1.pdf](http://www.ssi.gouv.fr/uploads/2015/01/RGS_v-2-0_B1.pdf)
- [18] [NIST SP 800-90A] NIST Released Special Publication (SP) 800-90A Revision 1: Recommendation for Random Number Generation Using Deterministic Random Bit Generators / National Institute of Standards and Technology (NIST), June 2015
- [19] [FIPS PUB 202] FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions / National Institute of Standards and Technology (NIST), August 2015
- [20] [FIPS PUB 198-1] FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION The Keyed-Hash Message Authentication Code (HMAC) / National Institute of Standards and Technology (NIST), July 2008
- [21] [SEC-recommended curves] SEC2: Recommended Elliptic Curve Domain Parameters, Certicom Research, v1.0, September 20, 2000