Common Criteria
Information Technology
Security Evaluation

Taurus1

STRONGV3P10_In04lpe of S5AV920/S5AV820/S5AV720 with Specific IC Dedicated Software, Version 2.0/2.1

Class: ASE

Version 0.1

2024-11-07

ST 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.

SAMSUNG ELECTRONICS

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。

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

Home Page: *http://www.samsungsemi.com*

Revision History

Revision No.	Date	Description
0.0	8th August 2024	Creation for initial version
0.1	7th November 2024	Update after CB Feedback

List of Figures

List of Tables

1 ST INTRODUCTION

1.1 Security Target Reference

Title:	Taurus1, STRONGV3P10_In04lpe of S5AV920/S5AV820/S5AV720 with Specific IC Dedicated Software, Version 2.0/2.1, ST (Security Target Lite)
Version:	0.1
Date:	2024-11-07
Developer:	Samsung Electronics Co., Ltd.
Certification ID:	NSCIB-CC-2300043-01

1.2 TOE Reference

The Target of Evaluation (TOE) is *STRONGV3P10_In04lpe of S5AV920/S5AV820/S5AV720 with specific IC Dedicated Software.* The TOE Version is *2.0* and *2.1.*

The TOE consists of the following components:

Table 1: TOE Components

Item type	Item	Version	Date	Format	Form of delivery
Hardware	STRONGV3P10_In04lpe TRONGV3P10	2.0	-	-	Hardware Secure Sub System as part of a SoC
Hardware	Secure JTAG Controller of SoC	2.0/2.1	-	-	Hardware as part of a SoC. Identified by SoC version[1].
Firmware	Secure Boot Loader	1.2	-	-	Stored in ROM of the STRONGV3P10_In04lpe
Software	AH0 Secure RSA/ECC/SHA Library (optional)	1.03	-	Binary File	It is optionally integrated into user code.
Software	DTRNG Library	1.2	-	Binary File	Encrypted email
Document	PKA Library API Manual (AH0 Secure RSA/ECC/SHA Library API Manual v1.05), Version 1.05, 2024-09-12, Samsung Electronics Co., Ltd.			PDF	Softcopy
Document	STRONG Secure Bootloader Manual for S5AV920, Revision 1.03, 2024-03-26, Samsung Electronics Co., Ltd.			PDF	Softcopy

[1] The complete TOE is identified by the SoC Version: *S5AV920/S5AV820/S5AV720 version 2.0 and 2.1.*

Item type	Item	Version	Date	Format	Form of delivery
Document	S5AV920 Chip Delivery Specification, Version 0.0, 2023-07-31, Samsung Electronics Co., Ltd.			PDF	Softcopy
Document	KITT2 HW DTRNG FRO M and DTRNG FRO M Library Application Note, Revision 1.2, 2024-03-26, Samsung Electronics Co., Ltd.			PDF	Softcopy
Document	STRONGV3P10 of S5AV920 32-bit RISC Microcontroller for Secure Element Platform User's Manual, Version 0.4, 2024-09-06 Samsung Electronics Co., Ltd.			PDF	Softcopy
Document	Security Application Note For STRONGV3P10 for S5AV920, Version 0.8, 2024-10-07, Samsung Electronics Co., Ltd.			PDF	Softcopy
Document	CORTEX-M35P Reference Manual, Version 0.0, 2020-05-11, ARM Limited			PDF	Softcopy
Document	Integration Guide, Version 0.6, 2024-10-07, Samsung Electronics Co., Ltd.			PDF	Softcopy

Note: Samsung is acting as the TOE developer as well as the 3S Integrator. The integration guidance [41] is therefore considered to be a Samsung-internal document which is not shared with the Composite Software Developer.

Table 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.

1.3 TOE Overview and TOE Description

1.3.1 TOE Type

The TOE is a Secure Sub-System (3S) implemented as a hard macro in a System on Chip (SoC) as defined in [PP, 1.2.1].

According to application note 3, the 3S is considered a monolithic IP block, but its functionality can be distributed across the SoC. In case of this TOE its functionality is split twofold in its core functionality and a Secure JTAG Controller to protect access to debug and test mode.

The TOE implements a processing unit, security components as well as volatile and non-volatile memories (hardware). The TOE also comprises dedicated firmware for loading the composite software from TOE-external memories and software components for cryptographic purposes and random number generation.

Any Composite Software is not part of the TOE. Security guidance for the Composite Software development is delivered as part of the TOE.

The TOE has dependencies on partial integration aspects, which are out of scope of the evaluated TOE.

1.3.2 TOE Definition

1.3.2.1 Physical Scope

The SoC S5AV920/S5AV820/S5AV720 has multiple CPUs as hosts. In addition, many peripheral IP blocks are connected via a multi-layered SoC interconnect. The TOE is implemented in form of a logical block of the SoC. A dedicated bridge is used to communicate with the SoC interconnect. The memories DRAM and NVM are located outside of the SoC.

The TOE hardware components are connected via a TOE-internal interconnect detailed in Figure 1.

The TOE hardware consists of the following components:

- 32-bit Central Processing Unit
- Memory Protection Unit (MPU)
- Memory Management Unit (MMU)
- ROM, SRAM, Crypto RAM
- OTP storage
- Secure JTAG Controller
- Detectors and security logic
- Random Number Generator
- AES cryptographic coprocessors
- Single-DES coprocessor
- Key manager
- Tornado-H arithmetic coprocessor
- SHA-2, SHA-3 and HMAC hardware engines
- Direct Memory Access
- Secure AXI Bridge
- Timers
- Mailboxes to communicate with the SoC main core
- DAP asynchronous bridge
- DMA
- APB Async_mi, APB2AHB, APB2AHB_remap
- SYS_REG
- APB4_slave_async_si, ALIVE_REG

The TOE further consists of the following firmware and software components:

- The Secure Boot Loader
- The DTRNG Library that fulfills the requirements of Class PTG.2 of BSI-AIS31 (German Scheme)
- The AH0 Secure RSA/ECC/SHA Library (optional) for extended cryptographic functionality.

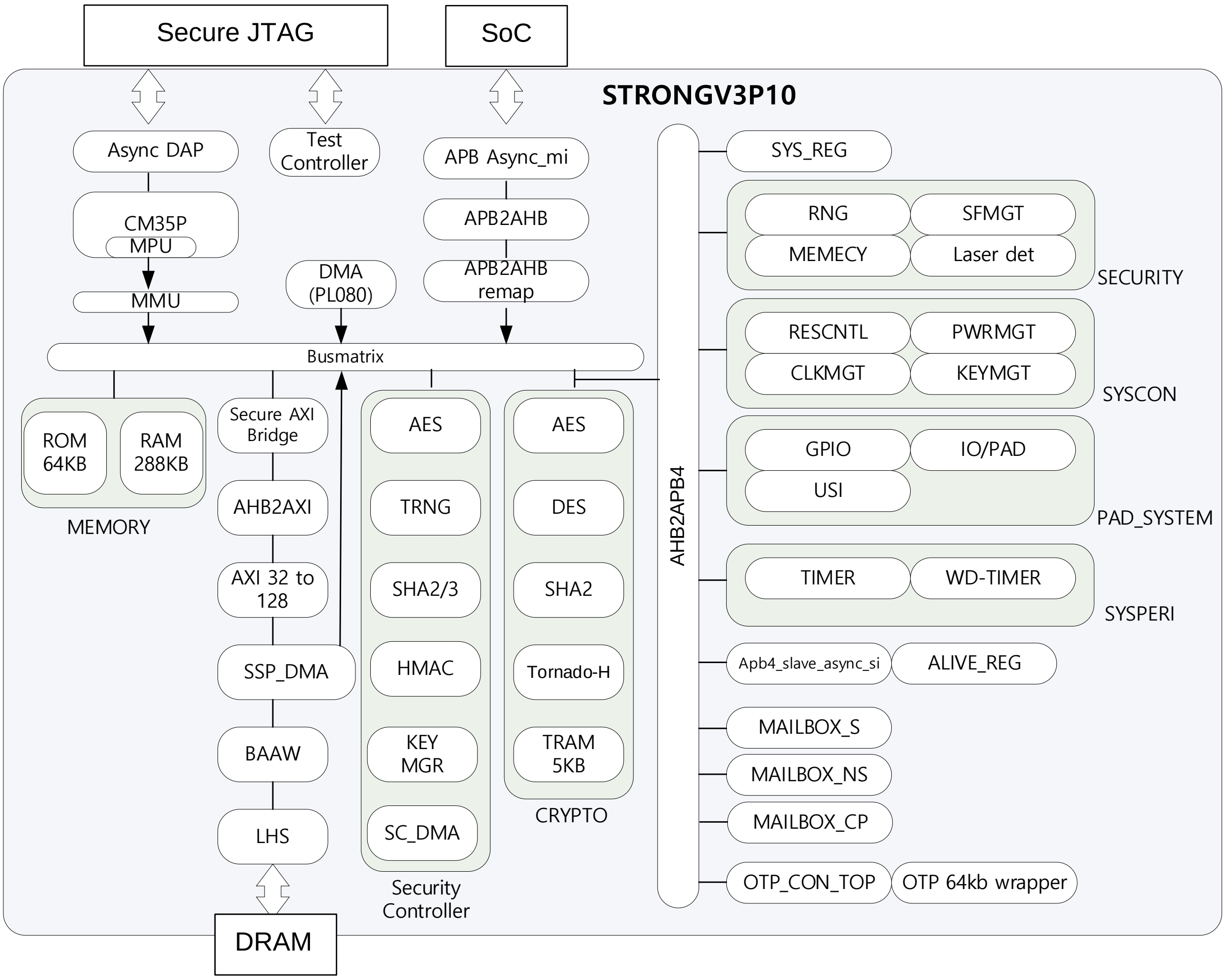

Figure 1: Block diagram of the TOE

The CORTEX-M35P CPU architecture of STRONGV3P10_In04lpe follows the Harvard architecture, which contains program and data memories. Using those separate memory access paths, both instruction and data can be fetched simultaneously without causing a stall.

Memory partitioning and access is protected by the MPU and MMU. The MPU supports the standard ARMv8 Protected Memory System Architecture (PMSA) and supports full support for protection regions, overlapping protection regions, access permissions, and exporting memory attributes to the system. The MPU can therefore be used to enforce privilege/access rules as well as to separate processes. The main purpose of the MMU is to map external memories like the DRAM to the internal memory. In case of address violations, the MMU raises an abort exception.

The JTAG Controller is used to prohibit access to and test mode until a successful authentication managed by the

JTAG Controller by comparison to a stored value in the OTP is executed.

The DMA (PL080) allows peripheral-to-memory, memory-to-peripheral, peripheral-to-peripheral, and memory-to-memory transactions. This functionality however is not in scope of the evaluation.

The (internal) mailbox interface of the TOE to the SoC is implemented via APB Async_mi, APB2AHB, APB2AHB remap.

The TORNADO-H is a coprocessor which is capable of performing big number arithmetic operations used for accelerating cryptographic operations. It aims to accelerate the implementation of public key cryptography algorithms such as RSA or ECC.

The TOE implements a PTG.2 True Random Number Generator in form of the DTRNG Library in conjunction with the TRNG block shown in Figure 1 as part of the Security Controller block.

The SYSCON block further implements features like reset control logic, power management, and clock management.

The SYSPERI block implements in timers and a watchdog timer.

The access to the external DRAM is implemented via the secure AXI bridge, SSP_DMA and LHS. Encryption and decryption of the DRAM content is done via the CRYPTO block (via SSP_DMA) or the Security Controller block (via SC_DMA) as controlled by the bootloader.
Please note that the execute in place functionality of the secure AXI bridge and SSP_DMA is not part of the TSF.

The OTP (one-time programmable) memory is a non-volatile memory that comes with a dedicated controller to manage the access to the memory. The OTP is used to store initialization and pre-personalization data of the TOE.

PP Application Note 1: The TOE is a hard macro as described in [PP, 1.2.2]. This ST adds additional SFRs to the TOE Security Functionality (TSF) to reflect the functionality of the TOE. Please further note that the TOE is only tested on the given SoCs and no portability of the evaluation results to other SoCs is intended.

PP Application Note 2: Interfaces with other SoC components are detailed in Section 1.3.2 and 1.4.

PP Application Note 3: The TOEs functionality is distributed twofold as described in section1.3.1.

PP Application Note 4: With respect to this application note, the Secure Boot Loader is considered firmware (FW), while the AH0 Secure RSA/ECC/SHA Library and Secure Boot Loader are considered Software (SW) (see also Table 1.

PP Application Note 5: Details regarding interfaces with other SoC components are provided in Section 1.3.2 and 1.4.

PP Application Note 6: See Section 1.4 for a description of TOE interfaces.

PP Application Note 7: Details regarding external interfaces are provided in Section 1.3.2 and 1.4.

PP Application Note 8: This ST uses the functional packages as identified in Section 2.3 satisfying the package dependencies described in the PP.

PP Application Note 9: Details regarding interfaces are provided in Section 1.3.2 and 1.4.

PP Application Note 10: The TOE has dependencies on the integration, which are outlined in the integration guide.

1.3.2.2 Logical Scope

1.3.2.2.1 Modes of Operation

The TOE implements the following modes of operation:

Table 3: Test, Debug, and Normal Mode

TEST mode	Debug mode	NORMAL mode
TEST mode of the TOE provides full access to the scan chain or memory/HM BIST logics.	Debug mode can be accessed through SoC secure JTAG.	Once the TOE is fused from TEST or Debug mode to NORMAL mode, it can no longer be brought back to TEST or Debug mode.

Figure 2: PRIVILEGE and USER Mode

Code can execute in PRIVILEGED or UNPRIVILEGED (user) mode. UNPRIVILEGED execution limits access to memory address spaces as configured by code executed in PRIVILEGED mode. PRIVILEGED execution has access to all resources. Transitions between these modes can be done via exceptions only.

1.3.2.2.2 Composite SW Loading

During start-up, the Secure Boot Loader is responsible to download, verify, and decrypt the composite software stored in external NVM to the internal SRAM. The bootloader further manages to transfer the current internal state to external DRAM in case of power-saving mode and to restore the content when leaving the power-saving mode.

The security features used by the Secure Boot Loader are mainly implemented in the Security Controller and CRYPTO blocks.

In case the composite software is loaded from an external NVM, the Secure Boot Loader first checks the integrity of the encrypted image with SHA-512. If there is no error, the proper key is taken from the OTP memory (default or OEM key). Before signature verification, message authentication is performed using HMAC. ECDSA is used to verify the signature and the FW version is checked to prevent roll-back. The encrypted image is finally decrypted with AES CBC and a final integrity check is performed on the decrypted image stored to internal SRAM. Afterwards, the Secure Boot Loader hands over to the composite software stored to the internal SRAM.

In case the back-up and restore functionality of the Secure Boot Loader is used, AES GCM with 256-bit keys is used for authenticated encryption/decryption of the SRAM contents to the external DRAM and vice versa. In this case, dedicated registers (ALIVE_REG in Figure 1) contain the backup DRAM address and size, the IV used for GCM encryption as well as a reference CRC32 value to check the integrity of these registers during start-up.

1.3.2.2.3 Cryptographic Functionality

The TOE hardware provides a co-processor implementing the single DES in ECB mode with three dedicated key registers that can be used to implement the Triple DES algorithm.

The TOE hardware further implements AES co-processors as part of the CRYPTO and Security Controller blocks supporting ECB, CBC, CTR, and GCM modes.

The TOE hardware also provides engines for

- SHA256/384/512 based on HASH standard-NIST FIPS 180-4
- SHA3 / SHAKE based on HASH standard-NIST FIPS PUB 202
- SHA2-based / SHA3-based HMAC

The AH0 Secure RSA/ECC/SHA Library (optional) makes use of the TORNADO-H arithmetic co-processor to implement the following features:

- RSA functionality:
  - RSA public/private key pair generation
  - RSA signature generation (standard and CRT)
  - RSA signature verification
  - R^2 value precomputation (standard and CRT)
- ECC functionality
  - Ephemeral or static key pair generation for ECDSA
  - ECDSA signature generation and verification for a message digest
  - ECDH secret key derivation
  - X25519 (DH with curve 25519)
  - X25519 with decoded scalar

  The library implements ECC for general curves over prime fields of sizes from 224 bits to 512 bits. Only cures the following curves are in scope of the certification:

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

The DTRNG Library makes use of the TRNG block shown in Figure 1 to implement a class PTG.2 random number generator of the German AIS31.

1.3.2.2.4 Security Measures

The implements the following security measures to protect its operation:

- Active shields against physical intrusive attacks
- Dedicated tamper-resistant design based on synthesizable glue logic and secure topology
- Dedicate hardware mechanisms against side-channel attacks, such as Random Branch Insertion in the

CPU or bus masking

- Dedicated hardware mechanisms against Fault Injection Attacks, such as redundancy (e.g. parity checks in CPU registers and memories, redundant hardware registers)
- Built-in circuits for resistance against side-channel and fault injection attacks in the single-DES and AES co-processors
- Abnormal condition detectors:
    - voltage, light, temperature sensors;
    - voltage glitch detectors (external and internal)
    - Active shield removal detector
    - Life time detector to check the integrity of the sensors and their signals
- Memory encryption and bus scrambling
    - Static bus scrambling
    - Dynamic data encryption/decryption of buses
    - Automatic ROM encryption/decryption
    - Automatic RAM encryption/decryption
- Integrity checks on memories and buses
    - ECC and parity calculators
    - 16-bit and 32-bit CRC for internal integrity protection

1.3.3 Usage and Major Security Features of a TOE

The major security features of the TOE are:

- Secure loading of customer software and secure backup from external memories
- Secure cryptographic functionality for a AES, RSA, ECDSA/ECDH, SHA-2/3, and HMAC
- Support to implement the TDES algorithm in the composite software in a secure fashion
- Secure AIS31-conformant random number generation (class PTG.2)
- Secure key storage in the OTP block and key manager to manage access to these keys
- Measures to protect the confidentiality and integrity of code and data while being processed by the TOE
- Measures to detect abnormal operating conditions (temperature, voltage, light)
- Measures to detect physical manipulation of the TOE (active shielding)

The TOE can be used in a wide range of applications like banking or automotive.

1.3.4 Required Non-TOE hardware/software/firmware

PP Application Note 11: Dependencies on the hosting SoC are outlined in the Integration Guide.

The TOE relies on external memories located outside of the SoC to store contents. Besides that, the TOE makes use of a SoC clock used for TOE-internal counters and to read OTP values before the internal clock stabilizes.

1.3.5 TOE Life cycle

With respect to the life-cycle model defined in the PP, the TOE is delivered after Phase 4 after packaging.

The development of the TOE and the SoC S5AV920/S5AV820/S5AV720 as well as the integration of the TOE to the SoC is completely and solely under control of Samsung. The following sites are included in the TOE's life-cycle up to delivery:

Table 4: Sites of the TOE life cycle

Site / Building	Name	Purpose	Phase
Hwasung Plant/ DSR Building	3S Firmware and Software Development	Firmware and Software development	Phase 2
Hwasung Plant/ DSR Building	3S Hardware development, 3S Integration in SoC	Design Center	Phase 2
Giheung Plant/ SR3 Building	3S Wafer Testing	Test program development	Phase 2
Hwasung Plant/ Line S3	3S in SoC Manufacturing	Wafer Fabrication	Phase 3
Hwasung Plant/ MR2 (NRD) Building	3S in SoC Manufacturing	Mask Fabrication/Shop (Preparation Room)	Phase 3
Giheung Plant/ Line 5	3S in SoC Manufacturing	Bump Fabrication	Phase 3
Giheung Plant/ Line 2	3S Wafer Testing, Initialization and Pre-personalization	EDS Test(Wafer Test)/Stock	Phase 3
TESNA Plant	3S Wafer Testing, Initialization and Pre-personalization	EDS Test (Wafer Test)	Phase 3
Onyang Plant/ Warehouse	3S in SoC Packaging	Packing/Warehouse/Delivery	Phase 4
Onyang Plant/ Line 2	3S in SoC Packaging	Stock/Grinding/Sawing /Packaging /Package Testing	Phase 3+4

PP Application Note 12: As the TOE firmware and software is developed by Samsung and the delivery is performed after Phase 4 of the life cycle defined in the PP, the split of the software development between Phase 1 and 2 is not relevant for the TOE. Therefore, this application note of the PP is not applicable.

PP Application Note 13: The integration guide is listed as a TOE component and assessed during AGD.

PP Application Note 14: The TOE is delivered after phase 4 of the life cycle model defined in the PP. At this point, trimming, initialization and pre-personalization are completed and the self-protection of the 3S is completely enabled.

PP Application Note 15: The TOE is delivered after phase 4 of the life cycle model defined in the PP. At this point, trimming, initialization and pre-personalization are completed and the self-protection of the 3S is completely enabled.

PP Application Note 16: The TOE is delivered after phase 4 of the life cycle model defined in the PP. At this point, all test and debug features are disabled. As further detailed in [36], the TOE supports one default public key during development and for key injection as well as three additional OEM keys. The hash values of the OEM keys are stored in the OTP block. One key is written by the developer and the other two can be fused by the customer before delivery of the final product.

1.4 Interfaces of the TOE

The TOE implements internal physical interfaces to the SoC and external physical interfaces to the external world.

- Debug Interface (external)
- Test Mode Interface (external)
- Supply Interface (external)
- SoC-specific Communication Interfaces (Mailbox Interface; internal)
- Interface towards external DRAM (internal)

The TOE further implements the following logical interfaces available to code executed on the TOE

- Special Function Registers
- CPU Instructions
- Interfaces of the Secure Boot Loader
- Interfaces of the DTRNG Library
- Interfaces of the AH0 Secure RSA/ECC/SHA Library

2 CONFORMANCE CLAIMS

2.1 CC Conformance Claim

This Security Target claims conformance to the Common Criteria Version 3.1 Revision 5:

- 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, see [1].
- 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, see [2].
- 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, see [3].

This Security Target is CC Part 2 extended and CC Part 3 conformant.

2.2 PP Claim

This Security Target is strictly conformant to the following protection profile (PP):

- Secure Sub-System in System-on-Chip (3S in SoC), Version 1.5, BSI-CC-PP-0117, see [5].

This Security Target does not claim conformance to any other protection profile.

PP Application Note 17: This application note is of explanatory nature only.

2.3 Package Claim

The assurance level for this Security Target is EAL5 augmented with AVA_VAN.5, ALC_DVS.2 and ALC_FLR.2.

This Security Target further claims conformance to the following functional packages defined in the PP:

- Package “Passive External Memory” conformant
- Package “Loader Functionality” conformant
- Package “Cryptographic Services” augmented

PP Application Note 43: All functional packages have been used in the ST only once, i.e. all iterations are reused from the PP without change.

2.4 Conformance Claim Rationale

The Evaluation Assurance Level (EAL) of the PP is EAL 4, augmented with the assurance components ATE_DPT.2, ALC_DVS.2, AVA_VAN.5 and ALC_FLR.2. The Assurance Requirements of the TOE obtain the Evaluation Assurance Level 5, augmented with the assurance components ALC_DVS.2, AVA_VAN.5 and ALC_FLR.2.

The Target of Evaluation (TOE) is a complete solution, implementing a Secure Sub-System (i.e. a hard macro in this case) as defined in the PP, section 1.2.2, so the TOE is consistent with the TOE type in the PP.

The security problem definition of this Security Target is consistent with the statement of the Security Problem Definition in the PP, as the Security Target claims strict conformance to the PP. Additional threats, organizational security policies and assumptions are introduced in Section 3 of this ST. A rationale is given in Section 4.

The Security Objectives of this Security Target are consistent with the statement of the Security Objectives in the PP, as the Security Target claims strict conformance to the PP. Additional Security Objectives are added in Section 4.1. A rationale given in Section 4.3.

The Security Requirements of this Security Target are consistent with the statement of the Security Requirements in the PP, as the Security Target claims strict conformance to the PP. Additional Security Requirements are added in Section 6. A rationale is given in Section 6.3. All assignments and selections of the security functional requirements are done in the PP and in Section 6 of this Security Target.

3 SECURITY PROBLEM DEFINITION

3.1 Assets

See PP, Chapter 3.1.

PP Application Note 18: The memories of the TOE provide very similar protective measures (memory encryption, address scrambling). Data stored in external memories is AES encrypted and its integrity is verified as part of the import procedure.

PP Application Note 19: This application note addressing wide-ranging protection mechanisms is of explanatory nature.

PP Application Note 20: The Secure Boot Loader is stored in ROM which is protected by a static memory encryption key and fixed address scrambling. The AH0 Secure RSA/ECC/SHA Library and DTRNG Library are integrated to the composite software and stored in the SRAM during operation of the TOE. The memory encryption and address scrambling used for SRAM is device-specific. When stored in external memories, the AH0 Secure RSA/ECC/SHA Library and DTRNG Library are either AES GCM encrypted (backup and restore functionality) or AES CBC encrypted and ECDSA signed (initial loading during start-up).

PP Application Note 44: The functional package “Passive External Memory” does not define additional assets.

PP Application Note 59: The functional package “Package for Loader Functionality” does not define additional assets.

3.2 Threats

The PP defines the following threats that are used in this ST without modification:

Table 5: Threats of the PP reused for the ST without modification

Threat	Title
Threats of the base PP defined in Chapter 3.2 of the PP	
T.Leak-Inherent	Inherent Information Leakage
T.Phys-Probing	Physical Probing
T.Malfunction	Malfunction due to Environmental Stress
T.Phys-Manipulation	Physical Manipulation
T.Leak-Forced	Forced Information Leakage
T.Abuse-Func	Abuse of Functionality

T.RND	Deficiency of Random Numbers
T.Insecure-State	Insecure State of the TOE
Threats of the functional package “Passive External Memory” defined in Chapter 7.1.1.2 of the PP	
T.Pas-Mem-Clone-Replace	Cloning or replacement of passive external memory
T.Pas-Mem-Content-Abuse	Abuse of passive external memory content
T.Pas-Mem-Cmd-Replay	Replay of commands between the 3S and the passive external memory
T.Pas-Mem-Unauth-Rollback	Unauthorised rollback of content in the passive external memory

In addition to the threats defined in the PP, this ST introduces the additional threat T.Mem-Access to cover intentional or unintentional attempts to access restricted memory areas containing either sensitive code or data. This can lead to the exposure or unauthorized modification of sensitive assets.

The TOE shall therefore avert the threat T.Mem-Access as specified below.

T.Mem-Access Memory Access Violation

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

PP Application Note 60: The functional package “Package for Loader Functionality” does not define additional threats.

3.3 Organizational Security Policies

The PP defines the following Organisational Security Policies that are used in this ST without modification:

Table 6: Organizational Security Policies reused for the ST without modification

Organizational Security Policy	Title
Organizational Security Policies of the base PP defined in Chapter 3.3 of the PP	
P.Gen-Unique-ID	Identification of each TOE instance
Organizational Security Policies of the functional package “Package for Loader Functionality” defined in Chapter 7.3.1.3 of the PP	
P.Access-Ctlr-Loader	Loader Functionality with User Authorisation
Organizational Security Policies of the functional package “Package for Cryptographic Services” defined in Chapter 7.4.1.3 of the PP	

P.Crypto-Service	Cryptographic service of the TOE

PP Application Note 45: The functional package “Passive External Memory” does not define additional Organisational Security Policies.

PP Application Note 65: The cryptographic services provided by the TOE are partly pure hardware features and partly implemented as a cryptographic library that makes use of a dedicated arithmetic coprocessor.

This ST does not add further Organizational Security Policy.

3.4 Assumptions

The PP defines the following Assumptions that are used in this ST without modification:

Table 7: Assumptions reused for the ST without modification

Assumption	Title
Assumptions of the base PP defined in Chapter 3.4 of the PP	
A.Process-Sec-IC	Protection during Packaging, Finishing and Personalisation
A.Resp-Appl.	Treatment of user data of the Composite Product

This ST does not add further Assumptions.

PP Application Note 21: The packaging of the SoC does not add further protection and is thus not included in the scope of the evaluation.

PP Application Note 22: The packaging of the SoC does not add further protection and is thus not included in the scope of the evaluation.

PP Application Note 23: The initialization of the TOE conducted by the bootloader and pre-personalisation containing the product specific software/firmware can be seen in Table 1 are part of the evaluated configuration.

PP Application Note 46: The functional package “Passive External Memory” does not define additional assumptions.

PP Application Note 61: The functional package “Package for Loader Functionality” does not define additional assumptions.

4 SECURITY OBJECTIVES

4.1 Security Objectives for the TOE

The PP defines the following Security Objectives that are used in this ST without modification:

Table 8: Security Objectives of the PP reused for the ST without modification

Security Objective	Title
Security Objectives of the base PP defined in Chapter 4.1 of the PP	
O.Leak-Inherent	Protection against Inherent Information Leakage
O.Phys-Probing	Protection against Physical Probing
O.Malfunction	Protection against Malfunction
O.Phys-Manipulation	Protection against Physical Manipulation
O.Leak-Forced	Protection against Forced Information leak
O.Abuse-Func	Protection against Abuse of Functionality
O.RND	Random Numbers
O.Secure-State	-
O.Identification	TOE Identification
Security Objectives of the functional package “Passive External Memory” defined in Chapter 7.1.2.1 of the PP	
O.Pas-Mem-Content-Prot	Protection against disclosure and undetected modification of passive external memory content
O.Pas-Mem-Cmd-Replay-Prot	Protection against replay of commands to store or modify data in passive external memory of the 3S
O.Pas-Mem-Unauth-Rollback-Prot	Protection against an unauthorised rollback of external memory content
O.Pas-Mem-Irreversible-Anchor	Passive external memory content Irreversibility Anchor
O.Pas-Mem-Clone-Replace-Prot	Protection against passive external memory cloning or replacement

Security Objective	Title
Security Objectives of the functional package “Package for Loader Functionality” defined in Chapter 7.3.2.1 of the PP	
O.Ctrl-Auth-Loader	Access control and authenticity for the Loader

In addition to the Security Objectives defined in the PP, this ST introduces the additional Security Objective O.Mem-Access to cover the threat T.Mem-Access.

O.Mem-Access Area based Memory Access Control

The TOE must provide the Composite 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.

PP Application Note 24: The TOE adds the Security Objective O.Mem-Access to counter the threat T.Mem-Access.

PP Application Note 66: As required by this application note, the security objective O.Crypto-Service shall list each cryptographic algorithm supported by the TOE.

O.Crypto-Service Cryptographic Algorithm

The TOE provides the cryptographic algorithm for the selected cryptographic operations and the selected modes of operation for the following Triple-DES, AES, RSA, ECC, SHA, HMAC, X25519.

4.2 Security Objectives for the Environment

The PP defines the following Security Objectives for the Operational Environment that are used in this ST without modification:

Table 9: Security Objectives for the Operational Environment of the PP reused for the ST without modification

Security Objective for the Operational Environment	Title
Security Objectives for the Operational Environment of the base PP defined in Chapter 4.2 of the PP	
OE.Resp-Appl	Treatment of user data of the Composite Product
OE.Secure-Initialisation	Uniqueness and authenticity of the device individual identifier
OE.Process-Sec-IC	Protection during Composite Product manufacturing
Security Objectives for the Operational Environment of the functional package “Package for Loader Functionality” defined in Chapter 7.3.2.2 of the PP	

Security Objective for the Operational Environment	Title
OE.Loader-Usage	Secure communication and usage of the Loader

PP Application Note 25: The packaging of the SoC does not add further protection and is thus not included in the scope of the evaluation.

PP Application Note 26: Package information is provided as part of [37].

PP Application Note 47: The functional package “Passive External Memory” does not define additional security objectives for the operational environment.

4.3 Security Objectives Rationale

In comparison to the PP only the threat T.Mem-Access and the objective O.Mem-Access are added to this Security Target.

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 Composite 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.

5 EXTENDED COMPONENTS DEFINITION

This Security Target reuses the extended components defined in the PP without modification. No further extended components have been added. Extended components are therefore defined in Chapter 5 and 7.1.3 of the PP.

PP Application Note 27: The Application Note is only of explanatory nature and the question, whether it is considered or not, is not applicable.

PP Application Note 28: The Application Note is only of explanatory nature and the question, whether it is considered or not, is not applicable.

PP Application Note 62: The functional package “Package for Loader Functionality” does not define extended components.

6 IT Security Requirements

In order to define the Security Functional Requirements the Part 2 of Common Criteria and the PP was used.

The operations of Security Requirements executed in this Security Target are described in this section. Please note that the following conventions are used to indicate these operations:

- Refinement operations are explicitly identified at the end of the security requirement definition.
- Assignment operations are identified in *italic*.
- Selection operations are identified by underline.
- Iterations are denoted using a slash "/".

In case security requirements are taken from the PP, this Security Target only highlights operations performed in this document.

6.1 Security Functional Requirements for the TOE

6.1.1 SFRs of the Base PP

The following SFRs are reused from the PP without modification.

Table 10: SFRs of the base PP reused for the ST without modification

SFR	Title
FRU_FLT.2/Env	Limited fault tolerance
FRU_FLT.2/Log	Limited fault tolerance
FPT_FLS.1/Env	Failure with preservation of secure state
FPT_FLS.1/Log	Failure with preservation of secure state
FMT_LIM.1/Test	Limited capabilities
FMT_LIM.2/Test	Limited availability
FMT_LIM.1/Debug	Limited capabilities
FMT_LIM.2/Debug	Limited availability
FPT_PHP.3	Resistance to physical attack

SFR	Title
FDP_ITT.1/3S	Basic internal transfer protection
FPT_ITT.1/3S	Basic internal TSF data transfer protection
FDP_IFC.1/3S	Subset information flow control

PP Application Note 29: This application note is of explanatory nature.

PP Application Note 30: The secure state is maintained by providing a non-maskable interrupt to the composite software.

PP Application Note 31: The secure state is maintained by providing a non-maskable interrupt to the Composite Software which is responsible to take appropriate actions.

PP Application Note 32: The TOE provides dedicated hardware registers (SECMON) that contain information about the error source responsible for triggering the non-maskable interrupt.

FDP_SDC.1/3S **Stored data confidentiality**

Hierarchical to: No other components.

Dependencies: No dependencies.

FDP_SDC.1.1/3S The TSF shall ensure the confidentiality of the information of the user data while it is stored in the *SRAM, TRAM, ROM, and OTP memory*.

FDP_SDI.2/3S **Stored data integrity monitoring and action**

Hierarchical to: FDP_SDI.1 Stored data integrity monitoring

Dependencies: No dependencies.

FDP_SDI.2.1/3S The TSF shall monitor user data stored in containers controlled by the TSF for *integrity errors* on all objects, based on the following attributes: *ECC, EDC, CRC, and parity errors in SRAM, TRAM, OTP memory read and execute operation*.

FDP_SDI.2.2/3S Upon detection of a data integrity error, the TSF shall enforce a *non-maskable interrupt (IRQ)*.

Refinement: This SFR applies for internal memory of the 3S.

PP Application Note 33: Internal memories of the TOE are SRAM, TRAM, ROM, and OTP.

PP Application Note 34: As an automatic response, the TSF creates a non-maskable interrupt towards the composite software which is then required to take further actions as described in the user guidance (see [40]).

FAU_SAS.1 **Audit storage**

Hierarchical to: No other components.

Dependencies: No dependencies.

FAU_SAS.1.1 The TSF shall provide the test process before TOE Delivery with the capability to store TOE unique identification data, Initialisation Data, Pre-personalisation Data in the *OTP memory*.

PP Application Note 35: The integrity and uniqueness of the unique identification of the TOE is supported by the development, production and test environment. This has been assessed in context of the evaluation activities.

PP Application Note 36: The data listed in FAU_SAS.1.1 is written to the OTP memory.

FPT_INI.1 **TSF Initialisation**

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_INI.1.1 The TOE initialization function shall verify correct configuration of configurable and/or trimmable security mechanisms and the unique identification, integrity of start-up software, correct initialisation of internal keys, *correct configuration of life cycle state such as security detectors and integrity checkers and bootloader secure sequence* prior to establishing the TSF in a secure initial state.

FPT_INI.1.2 The TOE initialization function shall detect and respond to errors and failures during initialization such that the TOE either successfully completes initialization or is halted.

FPT_INI.1.3 The TOE initialization function shall not be able to arbitrarily interact with the TSF after TOE initialization completes.

FCS_RNG.1/PTG.2 **Random number generation (Class PTG.2)**

Hierarchical to: No other components.

Dependencies: No dependencies.

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

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

(PTG.2.2) If a total failure of the entropy source occurs while the RNG is being operated, the RNG prevents the output of any internal random number that depends on some raw random numbers that have been generated after the total failure of the entropy source.

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

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

(PTG.2.5) The online test procedure checks the quality of the raw random number sequence. It is triggered applied upon specified internal events. The online test is suitable for detecting non-tolerable statistical defects of the statistical properties of the raw random numbers within an acceptable period of time.

FCS_RNG.1.2 The TSF shall provide numbers in the format *32-bit* that meet:

(PTG.2.6) Test procedure A does not distinguish the internal random numbers from output sequences of an ideal RNG.

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

PP Application Note 37: The ST uses the definition of the PTG.2 RNG class given in the German AIS31 scheme document [6]. The operations highlighted in FCS_RNG.1 above are the remaining operations left open in the AIS31 document.

6.1.2 SFRs of the Functional Package "Package for Passive External Memory"

PP Application Note 48: The iteration identifiers from the PP are reused for this ST without modification.

The following SFRs are reused from the functional package "Package for Passive External Memory" given in Chapter 7.1.4.1 of the PP without modification.

Table 11: SFRs of the functional package "Package for Passive External Memory" reused for the ST without modification

SFR	Title
FDP_DAU.2/PM	Data Authentication with Identity of Guarantor
FIA_UID.1/PM	Timing of identification
FDP_IRA.1/PM	Irreversibility Anchor for external memory
FDP_SDC.1/PM	Stored data confidentiality

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 passive 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.

PP Application Note 49: The TSF stops the boot procedure and returns an error message in case replay is detected.

FDP_URC.1/PM **Protection against an unauthorised rollback of memory content**

Hierarchical to: No other components.

Dependencies: FIA_UAU.1 or FDP_IRA.1

FDP_URC.1.1/PM The TOE shall detect an unauthorised replacement of the content stored in passive 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 unauthorised rollback of the content stored in a physically separated memory, the TOE shall stop TOE operation *and return an error status*.

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 *cryptographic integrity errors* on all objects, based on the following attributes: digital *signature*.

FDP_SDI.2.2/PM Upon detection of a data integrity error, the TSF shall *stop operation and return an error status*.

Refinement: This SFR applies for passive external memory.

6.1.3 SFRs of the Functional Package “Package for Loader Functionality”

The following SFRs are reused from the functional package “Package for Loader Functionality” given in Chapter 7.3.4.1 of the PP without modification.

Table 12: SFRs of the functional package “Package for Loader Functionality” reused for the ST without modification

SFR	Title
FDP_UCT.1/Load	Basic data exchange confidentiality
FDP_UIT.1/Load	Data exchange integrity

FTP_ITC.1/Load **Inter-TSF trusted channel**

Hierarchical to: No other components.

Dependencies: No dependencies.

FTP_ITC.1.1/Load The TSF shall provide a communication channel between itself and *the Composite*

Software Image Provider as described in the Loader SFP 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/Load The TSF shall permit another trusted IT product to initiate communication via the trusted channel.

FTP_ITC.1.3/Load The TSF shall initiate communication via the trusted channel for deploying *the bootloader for downloading the user data from external memory to internal SRAM*.

FDP_ACC.1/Load Subset access control - Loader

Hierarchical to: No other components.

Dependencies: FDP_ACF.1 Security attribute based access control.

FDP_ACC.1.1/Load The TSF shall enforce the Loader SFP on
(1) the subjects *Bootloader*,
(2) the objects user data in *internal SRAM in external memory*,
(3) the operation deployment of Loader.

PP Application Note 63: The Loader SFP is defined by FTP_ITC.1, FDP_UCT.1 and FDP_UIT.1 and FDP_ACF.1.

FDP_ACF.1/Load Security attribute based access control - Load

Hierarchical to: No other components.

Dependencies: FMT_MSA.3 Static attribute initialisation

FDP_ACF.1.1/Load The TSF shall enforce the Loader SFP to objects, based on the following:
(1) the subjects *Bootloader* with security attributes *Bootloader operating state and Bootloader key K_BL*
(2) the objects *user data in external memory* with security attributes *Software Image key K_I*.

FDP_ACF.1.2/Load The TSF shall enforce the following rules to determine whether an operation among controlled subjects and controlled objects is allowed:
(1) Bootloader changes the operating state from Bootloader State or Restore State to Firmware State only after successful verification of the authenticity of the user data downloaded from external memory to internal SRAM based on the Bootloader key K_BL and the Software Image key K_I.
(2) Bootloader changes the operating state from Firmware State to Restore State when copying the user data from internal SRAM to the external memory.

FDP_ACF.1.3/Load The TSF shall explicitly authorise access of subjects to objects, based on the following additional rules: None.

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

6.1.4 SFRs of the Functional Package “Cryptographic Services”

PP Application Note 67: As described in this application note of the PP, the SFR “FCS_COP.1/iteration” is replaced by the FCS_COP.1 iterations given in this Security Target.

PP Application Note 68: The set of cryptographic algorithms supported by the TOE is based on well established standards.

PP Application Note 69: As described in this application note of the PP, the SFR “FCS_CKM.4/iteration” is replaced by the FCS_CKM.4 iterations given in this Security Target.

6.1.4.1 Implemented in Hardware

FCS_COP.1/TDES **Cryptographic operation – TDES**

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/TDES The TSF shall perform *encryption and decryption* in accordance with a specified cryptographic algorithm *Triple Data Encryption Standard (TDES) in ECB mode* and cryptographic key sizes *112 bit and 168 bit* that meet the following: *NIST SP800-67 Rev. 2 [9] and NIST SP800-38A [11]*.

FCS_CKM.4/TDES **Cryptographic key destruction – TDES**

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4.1/TDES The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method *overwriting keys with zeros* that meets the following: *none*.

FCS_COP.1/AES **Cryptographic operation – AES**

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/AES The TSF shall perform *encryption and decryption* in accordance with a specified cryptographic algorithm *Advanced Encryption Standard (AES) in ECB, CTR, CBC, and GCM mode* and cryptographic key sizes *128 bit, 192 bit, and 256 bit* that meet the following: FIPS PUB 197 [10], NIST SP800-38A [11], and NIST SP800-38D [12].

FCS_CKM.4/AES **Cryptographic key destruction – AES**

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4.1/AES The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method *overwriting keys with zeros* that meets the following: *none*.

FCS_COP.1/KWP **Cryptographic operation – KWP**

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/KWP The TSF shall perform *key wrapping and unwrapping of AES keys* in accordance with a specified cryptographic algorithm *AES in KWP mode* and cryptographic key sizes *128 bit, 192 bit and 256 bit* that meet the following: FIPS PUB 197 [10] and NIST SP800-38F [13].

FCS_COP.1/KDF **Cryptographic operation – KDF**

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/KDF The TSF shall perform *cryptographic key derivation* in accordance with a specified cryptographic algorithm *KDF in counter mode with HMAC based on SHA2-256/384/512 and SHA3-224/256/384/512 as PRF* and cryptographic key sizes *256 bit, 512 bit, 768 bit, 1024 bit, and 1152 bit* that meet the following: NIST SP800-108 [14].

FCS_CKM.4/KDF **Cryptographic key destruction – KDF**

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4.1/KDF The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method *overwriting keys with zeros* that meets the following: *none*.

FCS_COP.1/SHA **Cryptographic operation – SHA**

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/SHA The TSF shall perform *secure hash computation* in accordance with a specified cryptographic algorithm *SHA2-256/384/512, SHA3-224/256/384/512, and SHAKE128/256* and cryptographic key sizes *none* that meet the following: FIPS PUB 180-4 [21] and FIPS PUB 202 [27].

FCS_COP.1/HMAC **Cryptographic operation – HMAC**

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/HMAC The TSF shall perform *keyed-Hash Message Authentication Code computation and verification* in accordance with a specified cryptographic algorithm *HMAC with SHA2-256/384/512, and SHA3-224/256/384/512* and cryptographic key sizes *256 bit, 512 bit, 768 bit, 1024 bit, and 1152 bit* that meet the following: FIPS PUB 198-1 [28].

FCS_CKM.4/HMAC **Cryptographic key destruction – HMAC**

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4.1/HMAC The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method *overwriting keys with zeros* that meets the following: *none*.

6.1.4.2 Implemented in the AH0 Secure RSA/ECC/SHA Library (optional)

6.1.4.2.1 Rivest-Shamir-Adleman (optional)

FCS_CKM.1/RSA **Cryptographic key generation – RSA (optional)**

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or
FCS_COP.1 Cryptographic operation]
FCS_CKM.4 Cryptographic key destruction

FCS_CKM.1.1/RSA The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *RSA key generation with and without CRT* and specified cryptographic key sizes *from 1900-bit up to 4096-bit with 2-bit granularity* that meet the following: *ETSI TS 102 176-1 [23], section 6.2.2.1 Key and parameter generation algorithm rsagen1 and ISO 18032 [16], Incremental search*.

FCS_COP.1/RSA **Cryptographic operation – RSA (optional)**

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/RSA The TSF shall perform the *modular exponentiation part of RSA signature generation and verification* in accordance with a specified cryptographic algorithm RSA and RSA-CRT and cryptographic key sizes *from 1900-bit up to 4096-bit with 2-bit granularity* that meet the following: *ISO/IEC14888-2:2008 [15], section 6.2 and 6.3*.

FCS_CKM.4/RSA **Cryptographic key destruction – RSA (optional)**

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4.1/RSA The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method *overwriting keys with zeros* that meets the following: *none*.

6.1.4.2.2 Elliptic Curve DSA (optional)

The AH0 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.

FCS_CKM.1/ECDSA Cryptographic key generation – ECDSA (optional)

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or
FCS_COP.1 Cryptographic operation]
FCS_CKM.4 Cryptographic key destruction

FCS_CKM.1.1/ECDSA The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *ECC* and specified cryptographic key sizes *from 224 bit up to 512 bit* that meet the following: *ANS X9.62 [17], Section A.4.3 Elliptic Curve Key Generation*.

FCS_COP.1/ECDSA Cryptographic operation – ECDSA (optional)

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/ECDSA The TSF shall perform *signature generation and verification* in accordance with a specified cryptographic algorithm *ECDSA* and cryptographic key sizes *from 224 bit up to 512 bit* that meet the following: *ANS X9.62 [17], Section 7.3 Signing Process and Section 7.4 Verifying Process*.

FCS_CKM.4/ECDSA Cryptographic key destruction – ECDSA

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4.1/ECDSA The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method *overwriting keys with zeros* that meets the following: *none*.

6.1.4.2.3 Elliptic Curve Diffie-Hellman (ECDH) Key Agreement (optional)

FCS_COP.1/ECDH Cryptographic operation – ECDH

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/ECDH The TSF shall perform *key agreement* in accordance with a specified cryptographic algorithm *ECDH* and cryptographic key sizes *from 224 bit up to 512 bit* that meet the following: *ANS X9.63 [18], Section 5.4.1 Standard Diffie-Hellman primitive*.

FCS_CKM.4/ECDH Cryptographic key destruction – ECDH

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4.1/ECDH The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method *overwriting keys with zeros* that meets the following: *none*.

6.1.4.2.4 X25519 (DH with curve25519) (optional)

FCS_COP.1/X25519 Cryptographic operation – X25519

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/X25519 The TSF shall perform *key agreement* in accordance with a specified cryptographic algorithm *X25519* and cryptographic key sizes *255 bits* that meet the following: *RFC 7748 [19]*.

6.1.4.2.5 Secure Hash Algorithm (optional)

The AH0 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 is required by the operating system. The SHA224, SHA256, SHA384 and SHA512 functionalities are intended to be used only for ECDSA signature generation and verification.

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 FIPS PUB 180-4.

FCS_COP.1/SHA_SW Cryptographic operation – SHA_SW

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/SHA_SW 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: FIPS PUB 180-4 [21].

6.1.5 Additional SFRs

6.1.5.1 Memory Access Control

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. This functionality is described as the following Memory Access Control Policy.

The security functional requirement “Subset access control (FDP_ACC.1/3S)” requires that this policy is in place and defines the scope were it applies. The security functional requirement “Security attribute based access control (FDP_ACF.1/3S)” 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.1/3S. 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.

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).

The Memory Access Control Policy defines the following objects, subjects, attributes, and operations.

Object	Description
Memory area(s)	The Memory Access Control Policy manages the access to memory areas located in SRAM, TRAM and Key Manager (containing OTP contents). Memory areas have to be defined by the composite software.

Subject	Description
Code running in User Mode	Any code running in User Mode.
Code running in Privileged Mode	Any code running in Privileged Mode.

Operation	Description
Read	Code running in any mode request read access to a specific object.
Write	Code running in any mode request write access to a specific object.
Execute	Code running in any mode request to execute code stored in a specific object.

Attribute	Description
Access control information	Each memory area is configures with read, write and execute access rights for code running in user mode and code running in privileged mode.

FDP_ACC.1/3S **Subset access control**

Hierarchical to: No other components.

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACC.1.1/3S The TSF shall enforce the *Memory Access Control Policy* on all *subjects, objects, and operations defined above*.

FDP_ACF.1/3S **Security attribute based access control**

Hierarchical to: No other components.

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

FDP_ACF.1.1 The TSF shall enforce the *Memory Access Control Policy* to objects based on the following: *read, write, and execute access rights of objects assigned to subjects*.

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 access control information before granting access so that accesses to be denied cannot be utilized 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*.

FMT_MSA.3 Static attribute initialization

Hierarchical to: No other components.

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

FMT_MSA.3.1 The TSF shall enforce the *Memory Access Control Policy* to provide *well-defined* default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 The TSF shall allow the *code running in privileged mode* to specify alternative initial values to override the default values when an object or information is created.

FMT_MSA.1 Management of security attributes

Hierarchical to: No other components.

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

FMT_MSA.1.1 The TSF shall enforce the *Memory Access Control Policy* to restrict the ability to change_default, modify, or delete the security attributes *access control information* to *code running in privileged mode*.

FMT_SMF.1 Specification of Management Functions

Hierarchical to: No other components.

Dependencies: No dependencies.

FMT_SMF.1.1 The TSF shall be capable of performing the following management functions: *access the control registers of the MPU*.

6.2 Security Assurance Requirements for the TOE

The Security Target will be evaluated according to

Security Target evaluation (Class ASE)

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

- Evaluation Assurance Level 5 (EAL5)

and augmented by taking the following components:

- ALC_DVS.2, AVA_VAN.5 and ALC_FLR.2.

corresponding to level “EAL5+”.

The assurance requirements are:

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)

Development Security	***(ALC_DVS.2)***
Flaw reporting procedures	***(ALC_FLR.2)***
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)**

PP Application Note 38: As required by this application note of the PP, this section defines the claimed SARs. The ST claims an augmented set of SARs to provide additional assurance to users of the TOE.

6.2.1 Refinements of the TOE Assurance Requirements

The refinements of ALC_DEL, ALC_CMC, ATE_COV, AGD_OPE, AGD_PRE and ADV_ARC are taken unchanged from [5], as the same component is used in this Security Target as in the claimed Protection Profile. The refinement from [5] of ALC_CMS has to be discussed in the Security Target as this document uses a higher component as the Protection Profile demands.

The refinement of ALC_CMS.4 from [5] can be applied to ALC_CMS.5 with aspects regarding the configuration control system for the TOE. The refinement is still valid.

PP Application Note 39: This ST does not introduce further SAR refinements. Therefore, this application note is not applicable.

PP Application Note 40: This application note has to be considered for the ALC evaluation of the TOE.

PP Application Note 41: This application note has to be considered for the ALC evaluation of the TOE.

PP Application Note 42: This application note has to be considered for the ALC evaluation of the TOE.

6.3 Security Requirements Rationale

6.3.1 Rationale for the SFRs

The rationale for the objectives and SFRs defined in the Protection Profile (chapter 6 and 7) are already described there and hence, will not be repeated in this Security Target. This chapter only contains a rationale for the newly defined objectives and SFRs.

Table 13: Security Requirements versus Security Objectives

Objective	TOE Security Functional and Assurance Requirements
O.Mem-Access	FDP_ACC.1/3S “Subset access control” FDP_ACF.1/3S “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.Crypto-Services	FCS_COP.1/TDES FCS_CKM.4/TDES FCS_COP.1/AES FCS_CKM.4/AES FCS_COP.1/SHA FCS_COP.1/KWP FCS_COP.1/KDF FCS_CKM.4/KDF FCS_COP.1/HMAC FCS_CKM.4/HMAC FCS_COP.1/RSA FCS_CKM.1/RSA FCS_CKM.4/RSA FCS_COP.1/ECDSA FCS_CKM.1/ECDSA FCS_CKM.4/ECDSA FCS_COP.1/ECDH FCS_CKM.4/ECDH FCS_COP.1/X25519 FCS_COP.1/SHA_SW

6.3.2 Dependencies of SFRs

The dependencies of the SFRs FDP_ITT.1/3S, FDP_IFC.1/3S, FPT_ITT.1/3S, FPT_PHP.3, FDP_SDC.1/3S, FRU_FLT.2/Env, FPT_FLS.1/Env, FRU_FLT.2/Env. FPT_FLS.1/Env, FRU_FLT.2/Log, FPT_FLS.1/Log, FDP_SDI.2/3S, FMT_LIM.1/Test, FMT_LIM.2/Test, FMT_LIM.1/Debug, FMT_LIM.2/Debug, FCS_RNG.1/PTG.2, FPT_INI.1, FAU_SAS.1, FDP_SDC.1/PM, FDP_SDI.2/PM, FPT_RPL.1/PM, FDP_IRA.1/PM, FDP_URC.1/PM, FDP_DAU.2/PM, FTP_ITC.1/Load, FDP_UCT.1/Load, FDP_UIT.1/Load, FDP_ACC.1/Load, FDP_ACF.1/Load are completely defined in the Protection Profile and hence, will not be repeated in this Security Target. This chapter only contains a rationale for the newly defined SFRs and the SFRs, whose dependency was not fulfilled in [5]. The text following the table discusses the remaining cases.

Table 14: Overview of SFR dependencies

Security Functional Requirement	Dependencies	Fulfilled by security requirements
FDP_ACC.1/3S	FDP_ACF.1	Yes, by FDP_ACF.1/3S
FDP_ACF.1/3S	FDP_ACC.1	Yes, by FDP_ACC.1/3S
	FMT_MSA.3	Yes, by FMT_MSA.3
FMT_MSA.1	FDP_ACC.1 or FDP_IFC.1	Yes, by FDP_ACC.1/3S
	FMT_SMR.1	Yes (by environment, see discussion below)
	FMT_SMF.1	Yes, by FMT_SMF.1
FMT_MSA.3	FMT_MSA.1	Yes, by FMT_MSA.1
	FMT_SMR.1	Yes (by environment, see discussion below)
FMT_SMF.1	No dependencies	-
FCS_COP.1/TDES	FCS_CKM.4	Yes
	FDP_ITC.1 or FDP_ITC.2 (if not FCS_CKM.1) or FCS_CKM.1	Yes (by environment, see discussion below)
FCS_CKM.4/TDES	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
	FDP_ITC.1 or FDP_ITC.2 (if not FCS_CKM.1) or FCS_CKM.1	Yes (by environment, see discussion below)
FCS_CKM.4/AES	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/KWP	FCS_CKM.4	Yes
	FDP_ITC.1 or FDP_ITC.2 (if not FCS_CKM.1) or FCS_CKM.1	Yes (by environment, see discussion below)

Security Functional Requirement	Dependencies	Fulfilled by security requirements
FCS_COP.1/KDF	FCS_CKM.4	Yes
	FDP_ITC.1 or FDP_ITC.2 (if not FCS_CKM.1) or FCS_CKM.1	Yes (by environment, see discussion below)
FCS_CKM.4/KDF	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	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
	FDP_ITC.1 or FDP_ITC.2 (if not FCS_CKM.1) or FCS_CKM.1	Yes (by environment, see discussion below)
FCS_CKM.4/HMAC	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
FCS_COP.1/RSA (optional)	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	Yes
	FCS_CKM.4	Yes
FCS_CKM.4/RSA (optional)	FDP_ITC.1 or FDP_ITC.2 (if not FCS_CKM.1) or FCS_CKM.1	Yes
FCS_COP.1/ECDSA (optional)	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	Yes
	FCS_CKM.4	Yes
FCS_CKM.1/ECDSA (optional)	FCS_COP.1 or FCS_CKM.2	Yes
	FCS_CKM.4	Yes
FCS_CKM.4/ECDSA (optional)	FDP_ITC.1 or FDP_ITC.2 (if not FCS_CKM.1) or FCS_CKM.1	Yes
FCS_COP.1/ECDH (optional)	FDP_ITC.1 or FDP_ITC.2, or FCS_CKM.1	Yes
	FCS_CKM.4	Yes
FCS_CKM.4/ECDH (optional)	FDP_ITC.1 or FDP_ITC.2 (if not FCS_CKM.1) or FCS_CKM.1	Yes
FCS_COP.1/X25519 (optional)	FCS_CKM.4	Yes
	FDP_ITC.1 or FDP_ITC.2 (if not FCS_CKM.1) or	Yes (by environment, see discussion below)

Security Functional Requirement	Dependencies	Fulfilled by security requirements
	FCS_CKM.1	
FCS_COP.1/SHA_SW (optional)	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1,FCS_CKM.4	See discussion below

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

The TOE provides cryptographic key generation only for RSA and ECDSA (FCS_CKM.1/RSA, FCS_CKM.1/ECDSA). The dependencies to cryptographic key generation (FCS_CKM.1) and import from outside of the TOE (FDP_ITC.1 and FDP_ITC.2) for other algorithms are not included into this Security target since the TOE only provide an engine for message digesting and encryption. However, the Composite Software may fulfil these requirements related to the needs of the implemented application. Thus, the dependent requirements concerning these functions shall be fulfilled by the environment (Composite Software).

The TOE provides cryptographic key destruction for X25519, but does not implement private key generation. However, the Composite Software may fulfil these requirements related to the needs of the implemented application. Thus, the dependent requirements concerning these functions shall be fulfilled by the environment (Composite Software).

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 the mentioned SFRs are not applicable for SHA and are therefore fulfilled.

6.3.3 Rationale for the Assurance Requirements

The PP [5] requires conformance to EAL4 augmented with ALC_DVS.2, ATE_DPT.2, AVA_VAN.5 and ALC_FLR.2. This ST claims conformance to EAL5 augmented with ALC_DVS.2, AVA_VAN.5 and ALC_FLR.2.

Table 15: SAR rationale

SARs of the PP [5]	SARs claimed in this ST	Comment
ADV_ARC.1	ADV_ARC.1	Consistent.
ADV_FSP.4	ADV_FSP.5	The requirements of ADV_FSP.5 requires a complete semi-formal description with additional error information to provide advanced insight to the TOE functionality and behaviour for the evaluation.
ADV_IMP.1	ADV_IMP.1	Consistent.
–	ADV_INT.2	ADV_INT.2 requires well-structured internals. This ST claims the SAR to improve the security and testability of the TOE.
ADV_TDS.3	ADV_TDS.4	The requirements of ADV_TDS.4 requires a semi-formal modular design to provide advanced insight to the TOE

SARs of the PP [5]	SARs claimed in this ST	Comment
		functionality, behaviour, and internal processes for the evaluation.
AGD_OPE.1	AGD_OPE.1	Consistent.
AGD_PRE.1	AGD_PRE.1	Consistent.
ALC_CMC.4	ALC_CMC.4	Consistent.
ALC_CMS.4	ALC_CMS.5	ALC_CMS.5 requires to also track development tools and related information in the CM system. This guarantees the desired quality standards for the development of the TOE.
ALC_DEL.1	ALC_DEL.1	Consistent.
ALC_DVS.2	ALC_DVS.2	Consistent.
ALC_FLR.2	ALC_FLR.2	Consistent.
ALC_LCD.1	ALC_LCD.1	Consistent.
ALC_TAT.1	ALC_TAT.2	ALC_TAT.2 requires the developer to provide implementation standards used by the developer. Similar to ALC_CMS.5, this increased SAR components is used to represent the quality standards for the development of the TOE.
ASE_CCL.1	ASE_CCL.1	Consistent.
ASE_ECD.1	ASE_ECD.1	Consistent.
ASE_INT.1	ASE_INT.1	Consistent.
ASE_OBJ.2	ASE_OBJ.2	Consistent.
ASE_REQ.2	ASE_REQ.2	Consistent.
ASE_SPD.1	ASE_SPD.1	Consistent.
ASE_TSS.1	ASE_TSS.1	Consistent.
ATE_COV.2	ATE_COV.2	Consistent.
ATE_DPT.2	ATE_DPT.3	ATE_DPT.3 requires to test against the modular design of ADV_TDS. The ST claims the increased SAR component to provide further assurance in the correct functionality of the TOE.

SARs of the PP [5]	SARs claimed in this ST	Comment
ATE_FUN.1	ATE_FUN.1	Consistent.
ATE_IND.2	ATE_IND.2	Consistent.
AVA_VAN.5	AVA_VAN.5	Consistent.

7 TOE SUMMARY SPECIFICATION

7.1 List of Security Functional Requirements

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

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 enables the composite software to take appropriate actions.

The state is monitored by the TOE's detectors. The TOE's detectors detect abnormal voltage supply, abnormal environmental temperature and power glitches that are out of the specified range (refer to chapter 1.3.3). If the failures are happening, the TOE goes into IRQ state i.e., triggering an exception which must be handled by the composite software. This satisfies the FPT_FLS.1/Env "Failure with preservation of secure state". For more information, refer to the following description of the TOE's detectors.

FPT_FLS.1/Log

The detection thresholds of TOE's detectors are inside the operating range of the TOE. Therefore, abnormal interface behaviour and/or protocol parameters or protocol sequences are detected before the secure state is compromised. This enables the composite software to take appropriate actions.

The state is monitored by the TOE's detectors. The TOE's detectors are monitoring the failure occurs. The failures are abnormal voltage, abnormal temperature, and power glitch detectors that detect out of the specified range (refer to chapter 1.3.3). If the failures are happening, the TOE goes into IRQ state. This satisfies the FPT_FLS.1/Log "Failure with preservation of secure state". For more information, refer to the following TOE's Detectors.

TOE's Detectors

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

- The TOE hardware shall be configures to generate non-maskable interrupts (IRQ) when an event is detected and logs the event to a special function register (SECMON)

List of detectors:

- Abnormal voltage Detector
- Abnormal temperature Detector
- Laser Detector
- Active shield removal Detector
- External/Internal Glitch Detector
- Life Time Detector

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

SFR2: FRU_FLT.2: Limited fault tolerance

FRU_FLT.2/Env

These Integrity Checkers are used for preventing noise and laser from causing undefined or unpredictable behaviour of the chip.

- Partiy bits for bus and CPU data registers and DMA
- ECC for TRAM
- EDC for ROM and SRAM
- CRC/EDC/ECC for OTP
- CRC32 Calculator
- Dual Flip Flops for security relevant SFRs

FRU_FLT.2/Log

These Integrity Checkers are used for preventing noise and laser from causing abnormal interface behaviour and/or protocol parameters or protocol sequences.

- Parity for bus, CPU data registers and DMA control registers
- Tag of AES GCM mode used by the secure DMA(SC_DMA) during restore/backup-operation

SFR3: FPT_PHP.3: Resistance to physical attacks

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.

Static Address/Data scrambling for bus and memory protects memory and address/data bus from probing attacks.

Synthesizable processor core: The Central Processing Unit (CPU) of the TOE is synthesizable with glue logic, which makes reverse engineering and signal identification more difficult. Most sensitive hardware components such as buses are also hidden and implemented in deepest layers.

SFR4: FDP_ACC.1/3S: Subset access control

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

1) Security registers access control:

2) Invalid address access:

3) Access rights for the code executed in SRAM:

4) Communication between TOE and the external memories via secure DMA (SC_DMA) :

5) Protection of TOE against entities outside of the TOE:

SFR5: FDP_ACF.1/3S: Security attributes based access control.

This is covered by the Privilege and User modes of the TOE. For more information, refer to Figure 2. Privilege and User Modes.

SFR6: FMT_MSA.3: Static attribute initialization.

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

The access attribute of ROM and SRAM memory have DEFAULT values: Read-only/execute attribute for ROM and Read Write Non-execute attribute for SRAM.

SFR7: FMT_MSA.1: Management of security attributes.

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. This allows the operating system to control the memory regions accessible by a User mode application process. The protection unit enables user to divide memory into 8 regions, each with their own access permission attributes. If access against the set condition is performed, chip automatically generates IRQ.

SFR8: FMT_SMF.1: Specification of management functions.

This is achieved via access to Special Function Registers of Memory Protection Unit (MPU). MPU provides Special Function Registers which defines the base address and the limit address for a partition. The Registers exist for SRAM, and TRAM. Additional Registers exist for defining the protection attribute for each partition.

SFR9: FAU_SAS.1: Audit Storage

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

1) Non-reversibility of TEST mode and NORMAL mode:
2) TEST mode communication protocol and data commands:
3) Functional Tests:.
4) Identification:

SFR10: FMT_LIM.1: Limited capabilities

FMT_LIM.1/Test

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

FMT_LIM.1/Debug

Debug mode can be accessed only by the Debugger in Debugging step. Once the TOE is changed to NORMAL mode, Debug mode functions are no more available.

SFR11: FMT_LIM.2: Limited availabilities

FMT_LIM.2/Test

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

FMT_LIM.2/Debug

Debug mode can only be accessed in if the TOE is in debug mode. Once the TOE is changed to NORMAL mode, Debug mode commands are no more available. Debugging is realized through JTAG to enable composite software development.

SFR12: FDP_IFC.1/3S: Subset information flow control

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. The algorithms used for encryption are proprietary. The ROM encryption is static key while the RAM encryption is dynamic key. RAM encryption is performed automatically.

Life time detector: The life time detector is a special circuitry to check the integrity of the remaining detectors and their signals.

SFR13: FDP_ITT.1/3S: Basic internal transfer protection

This requirement is achieved by the combination of the TOE security features TOE features 1) to 5) to protect internal signals from disclosure and manipulation.

1) Static Address/Data scrambling for bus and memory: This function protects memory and address/data bus from probing attacks.
2) Dynamic Data encryption for bus: This function protects data bus from probing attacks.
3) Memory encryption: This security function protects the memory contents of the TOE from data analysis on the stored data . The algorithms used for encryption are proprietary.
4) Synthesizable processor core: The Central Processing Unit (CPU) of the TOE is synthesized in glue logic, which makes reverse engineering and signal identification more difficult. Most sensitive hardware components such as buses are also hidden and implemented in deepest layers.
5) De-synchronization and signal-to-noise ratio reduction mechanisms: The TOE operations can be made asynchronous by using the Random Branch insertion security features as configures by the Composite Software. They make a full range of intrusive (e.g. probing attacks) and non-intrusive attacks (e.g. side-channel attacks) more complex and difficult.

SFR14: FPT_ITT.1/3S: Basic internal TSF data transfer protection

See SFR13.

SFR15: FCS_RNG.1/PTG.2: Random number generation

This requirement is ensured by the design of the random number generation algorithm that makes use of Digital True Random Number Generator (DTRNG FRO M) and the associated DTRNG FRO M library conforming to *BSI-*

AIS-31 Class PTG.2 requirements (German scheme).

SFR16: FCS_COP.1: Cryptographic operation

This requirement is covered by the TOE.

DES Data Encryption Standard Engine with TDES implemented in software

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)

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)

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)

This function supports to calculate hash (digest) values. (FCS_COP.1/SHA)

As the hash functions do not implement measures countering leakage or fault-injection attacks, these functions must not be used to hash security values like keys etc.

HMAC (Keyed-Hash Message Authentication Code)

This function supports to calculate keyed-hash (digest) values. (FCS_COP.1/HMAC)

TORNADO-H RSA Cryptographic Library as part of AH0 Secure RSA/ECC/SHA library (optional)

This function assists in the acceleration of modulo exponentiations required in the RSA encryption/decryption algorithm. (FCS_COP.1/RSA)

TORNADO-H is a high speed modular multiplication coprocessor for the support of the RSA public key cryptosystem. The optional AH0 Secure RSA/ECC/SHA library is the software built on the TORNADO-H coprocessor that provides high level interface for RSA-based algorithms.

The TND_RSA_SigSTD_Secure and TND_RSA_SigCRT_Secure have some countermeasure against the timing attack, SPA, DPA and the fault attack.

The RSA_R2modM_precompute_sec and RSA_R2modPandQ_precompute_sec functions implement some countermeasures against the fault attack.

TORNADO-H ECC Cryptographic Library as part of AH0 Secure RSA/ECC/SHA library (optional)

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)

AH0 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.

The functions ECDSA_sign_digest and ECDH_generate have some countermeasure against the timing attack, SPA and the fault attack. The ECDSA_verify_digest function has some countermeasures against the fault attack.

The AH0 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 180-4], but only the functions related to SHA224, SHA256, SHA384 and SHA512 listed below are in the scope of this evaluation (FCS_COP.1/SHA_SW):

- SHA224_init, SHA224_update, SHA224_final,
- SHA256_init, SHA256_update, SHA256_final.
- SHA384_init, SHA384_update, SHA384_final.
- SHA512_init, SHA512_update, SHA512_final.

As the hash functions do not implement measures countering leakage or fault-injection attacks, these functions must not be used to hash security values like keys etc. The TOE provides the functionality of hash computation if and only if the optional AH0 Secure RSA/ECC/SHA library is delivered.

SFR17: FCS_CKM.1: Cryptographic key generation

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

- RSA_KeyGen_Secure - FCS_CKM.1/RSA.
- ECDSA_keygen - FCS_CKM.1/ECDSA.

SFR18: TSF Initialisation (FPT_INI.1)

This requirement is achieved by correct configuration of life cycle state. For more information refer to the following information.

1) Security detectors such as Abnormal voltage Detector, Abnormal temperature Detector, Laser Detector, Active shield removal Detector, External/Internal Glitch Detector, Life Time Detector
2) Integrity Checker such as parity and ECC
3) Bootloader sequence such as Integrity check and ECDSA sign and verification

SFR19: Reserved

SFR20: Inter-TSF trusted channel (FTP_ITC.1/Load)

This requirement is achieved by processing the authentication sequence of the bootloader.

This channel is distinct from other communication channels and provides assured identification for its end points and protection of the channel data from modification or disclose by ECDSA sign and verification sequence.

SFR21: Basic data exchange confidentiality (FDP_UCT.1/Load)

This requirement is achieved by secure external FLASH loading. User data which is loaded from the external FLASH memory is encrypted data. All data are encrypted and decrypted using AES algorithm.

SFR22: Data exchange integrity (FDP_UIT.1/Load)

This requirement is achieved by checking the checksum. Bootloader supports the checksum operation.

SFR23: Subset access control - Loader (FDP_ACC.1/Load)

This requirement is achieved by following functions.

Access attribute control of Bootloader:

SRAM memory attribute as “Disable code execution on RAM”: The attribute of SRAM is changed from “Enable code execution on RAM” to “Disable code execution on RAM”. If there is an unallowable access, hard fault interrupt is generated.

SFR24: Security attribute based access control - Load (FDP_ACF.1/Load)

This is covered by the ROM Booting and SRAM Booting mode of the TOE. TOE can be set to ROM Booting and SRAM Booting mode domains exclusively. The encryption key for SRAM loading of Bootloader is accessible only in ROM Booting mode. The SRAM Booting mode cannot access the encryption key for SRAM loading of Bootloader.

SFR25: Stored data confidentiality (FDP_SDC.1/3S)

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.

SFR26: Stored data integrity monitoring and action (FDP_SDI.2/3S)

This requirement is achieved by following functions.

SRAM ECC: Each 32-bit data/code in SRAM is associated with 7bit Error-Correcting-Code. It is always activated, and each time data/code is read from SRAM, the ECC is also read, and verified. In the verification, the ECC is calculated from data/code and modified with the value in single bit error. If the computed and read values are double bit different, it invokes an interrupt.

TRAM parity: TRAM data has parity check. If this check fails, the device generates an interrupt.

SFR27: Reserved

SFR28: Data Authentication with Identity of Guarantor (FDP_DAU.2/PM)

The image of the FW is stored with a certificate and the TOE stores a means to verify the public key. The authenticity is checked by using the received message as a message digest that is digitally signed.

SFR29: Timing of identification (FIA_UID.1/PM)

The identification of the encrypted software image is done before decrypting the image. The authenticity is checked by verification of an HMAC and digital signature stored within the encrypted image.

SFR30: Replay detection (FPT_RPL.1/PM)

After downloading the image (containing the firmware) from external Passive NVM the integrity is verified with a SHA512 (SHA-2/ SHA-3 hardware engines in the Security Controller), and next the signature is verified. The mechanism used for the signature verification will detect replay errors in the reading, writing and erasing commands that the 3S has issued to the external NVM. If the verification fails, the TOE will not boot and return an error.

SFR31: Protection against an unauthorized rollback of memory content (FDP_URC.1/PM)

A set of rollback counters implemented in OTP prevent the TOE from loading an older version of firmware.

The encrypted TAG that is maintained with the in DRAM stored firmware, is tied to the IV of the AES GCM that is randomly generated and stored inside the TOE every time the firmware is stored externally. This mechanism prevents unauthorized rollback as part of the backup and restore functionality of the Bootloader.

SFR32: Irreversibility Anchor for external memory (FDP_IRA.1/PM)

The TOE implements a number of rollback counters in OTP that check at booting the freshness of the firmware stored encrypted in the external Passive NVM and prevent the loading of older versions of the firmware.

When user data is uploaded to the DRAM , it is stored encrypted under GCM with a TAG tied to specific IV that stays within the TOE, This is used to prevent rollback and determines the freshness of the stored data.

SFR33: Stored data confidentiality (FDP_SDC.1/PM)

The image (containing the firmware) that can be downloaded from the external passive NVM is stored encrypted using AES CBC in the external passive NVM.

During operation of the TOE the firmware can be temporarily uploaded to and downloaded from DRAM. The temporarily stored firmware is stored encrypted using AES GCM.

SFR34: Stored data integrity monitoring and action (FDP_SDI.2/PM)

The integrity of the firmware stored in external passive memory is verified twice. The first time upon downloading the image containing the firmware and the second time after decryption and verifying the authenticity of the content of the image so a check on the integrity of the firmware itself.

During operation the integrity during uploading and downloading of the firmware to the DRAM is protected by the AES GCM that is used for encryption and decryption of the firmware.

SFR35: FCS_CKM.4: Cryptographic key destruction

This requirement is covered by the TOE.

Cryptographic Key destruction – Triple Data Encryption Standard Engine

This requirement is achieved by overwriting the TDES key register(FCS_CKM.4/TDES).

Cryptographic Key destruction – AES (Advanced Encryption Standard)

This requirement is achieved by “overwriting the AES key register(AES in CRYPTO block)” or by “setting AES key clear bit of AES control register(AES in Security Controller block)” (FCS_CKM.4/AES).

Cryptographic Key destruction – KDF (Key Derivation Function, Key Manager)
This requirement is achieved by setting the KDF key clear bit of KDF control register (FCS_CKM.4/KDF).

Cryptographic Key destruction – HMAC (Keyed-Hash Message Authentication Code)
This requirement is achieved by setting the HMAC key clear bit of HMAC control register (FCS_CKM.4/ HMAC).

Cryptographic Key destruction – RSA (Rivest-Shamir-Adleman)
This requirement is achieved by overwriting keys stored by the library in crypto. RAM and/or RAM (FCS_CKM.4/RSA).

Cryptographic Key destruction – ECDSA (Elliptic Curve Digital Signature Algorithm)
This requirement is achieved by overwriting keys stored by the library in crypto. RAM and/or RAM(FCS_CKM.4/ECDSA).

Cryptographic Key destruction – ECDH (Elliptic Curve Diffie-Hellman)
This requirement is achieved by overwriting keys stored by the library in crypto. RAM and/or RAM(FCS_CKM.4/ECDH).

8 Annex

8.1 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] Secure Sub-System in System-on-Chip (3S in SoC), Version 1.5, BSI-CC-PP-0117

[6] AIS31: Functionality classes and evaluation methodology for physical random number generators, Version 3, 15.05.2013, 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] NIST SP 800-67 Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher, Revision 2

[10] FIPS PUB 197 Advanced Encryption Standard (AES), 2001-11-26, National Institute of Standards and Technology.

[11] NIST SP 800-38A Recommendation for Block Cipher Modes of Operation – Methods and Techniques, December 2001, National Institute of Standards and Technology.

[12] NIST SP 800-38D Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC, November 2007, National Institute of Standards and Technology.

[13] NIST SP 800-38F Recommendation for Block Cipher Modes of Operation: Methods for Key Wrapping, December 2012, National Institute of Standards and Technology.

[14] NIST SP800-108 Recommendation for Key Derivation Using Pseudorandom Functions, October 2009, National Institute of Standards and Technology.

[15] ISO/IEC 14888-2:2008 - Information technology -- Security techniques-- Digital signatures with appendix -- Part 2: Integer factorization based mechanisms.

[16] ISO/IEC 18032:2020 – Information security – Prime number generation, 2020-12.

[17] American National Standard for Financial Services ANS X9.62-2005, Public Key Cryptography for the Financial Services Industry, The Elliptic Curve Digital Signature Algorithm (ECDSA), November 16, 2005, American National Standards Institute.

[18] American National Standard for Financial Services X9.63-2011, Public Key Cryptography for the Financial Services Industry - Key Agreement and Key Transport Using Elliptic Curve Cryptography, December 21, 2011, American National Standards Institute.

[19] RFC 7748 – Elliptic Curves for Security, January 2016, Internat Research Task Force (IRTF), https://datatracker.ietf.org/doc/html/rfc7748.

[20] Supporting Document: Application of Attack Potential to Smartcards June 2020, Version 3.1.

[21] FIPS PUB 180-4 U.S. Department of Commerce / National Bureau of Standards, Secure Hash Algorithm, FIPS PUB 180-4, 2008-October

[22] FIPS PUB 186-4 Digital Signature Standard, June 2009, National Institute of Standards and Technology.

[23] 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

[24] T. Finke, M. Gebhardt and W. Schindler, A New Side-Channel Attack on RSA Prime Generation, CHES 2009, LNCS 5747, pp. 141-155, 2009.

[25] 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

[26] NIST SP 800-90A Recommendation for Random Number Generation Using Deterministic Random Bit Generators, June 2015, National Institute of Standards and Technology.

[27] FIPS PUB 202 SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions, August 2015, National Institute of Standards and Technology.

[28] FIPS PUB 198-1 The Keyed-Hash Message Authentication Code (HMAC), July 2008, National Institute of Standards and Technology (NIST).

[29] SEC2: Recommended Elliptic Curve Domain Parameters, v1.0, September 20, 2000, Certicom Research.

[30] Supporting Document: The Application of CC to Integrated Circuits, March 2009, Version 3.0,Revision 1

[31] Supporting Document Guidance: Smartcard Evaluation, February 2010, Version 2.0

[32] Supporting Document: Security Architecture requirements (ADV_ARC) for smart cards and similar devices, Jan 2012, Version 2.0

[33] Supporting Document: Composite product evaluation for Smart Cards and similar devices, May 2018, Version 1.5.1

[34] Supporting Document: Minimum site security requirements, Feb. 2020, Version 3.0

[35] PKA Library API Manual (AH0 Secure RSA/ECC/SHA Library API Manual v1.05), Version 1.05, 2024-09-12, Samsung Electronics Co., Ltd.

[36] STRONG Secure Bootloader Manual for S5AV920, Revision 1.03, 2024-03-26, Samsung Electronics Co., Ltd.

[37] S5AV920 Chip Delivery Specification, Version 0.0, 2023-07-31, Samsung Electronics Co., Ltd.

[38] KITT2 HW DTRNG FRO M and DTRNG FRO M Library Application Note, Revision 1.2, 2024-03-26, Samsung Electronics Co., Ltd.

[39] STRONGV3P10 of S5AV920 32-bit RISC Microcontroller for Secure Element Platform User’s Manual, Version 0.4, 2024-09-06, Samsung Electronics Co., Ltd.

[40] Security Application Note For STRONGV3P10 for S5AV920, Version 0.8, 2024-10-07, Samsung Electronics Co., Ltd.

[41] Taurus1 Integration Guide, Version 0.6, 2024-10-07, Samsung Electronics Co., Ltd.