Titan
H1D3 Secure Microcontroller with Crypto Library v1.3.10 Security Target Lite

Version: 4.5
Release: Sep 25, 2023

1 Introduction

1.1 ST reference

Titan H1D3 Secure Microcontroller with Crypto Library v1.3.10 Security Target Lite, Revision 4.5, Google LLC, September 25th, 2023.

1.2 TOE reference

The TOE is named “H1D3 Secure Microcontroller with Crypto Library v1.3.10”. It consists of:

- The Secure Microcontroller H1D3
- IC Dedicated Software
    - Bootloader stored in ROM
    - Crypto Library v1.3.10 that is to be loaded into Flash together with the Embedded Software
- Documentation describing usage of the TOE

For more details on the identification of the different components please refer to Section 1.6.

1.3 Purpose

The TOE is used in smartphone, servers and personal computers to increase the security of the platform including but not limited to secure boot, user authentication, and user data protection.

1.4 Conventions

None

1.5 TOE Overview

1.5.1 TOE definition

The H1D3 Secure Microcontrollers are provided in one of three packages referenced H1D3M, H1D3C and H1D3P. The Secure Microcontroller is based on a flash-based secure microcontroller platform. A RISC-V core named Soteria alongside RAM, ROM and flash memories and cryptographic hardware accelerators provides the root to run secure applications. The TOE includes a Crypto Library.
The image loaded and verified by the TOE bootloader stored in ROM includes the Crypto Library in addition to the Embedded Software. The usage of the TOE consists of:

- Developing IC Embedded Software that uses the security services provided by the TOE
- Loading the IC Embedded Software using the bootloader.

- Executing the IC Embedded Software

Sections 1.5.2 and 1.5.3 describe the major security features of the hardware and software parts of the TOE, respectively

1.5.2 Hardware

1.5.2.1 TOE boundary

The Hardware part of the TOE includes the following components:

- RISC-V Soteria CPU with a single execution mode
- Memory Protection Unit (MPU)
- Flash memory with two banks (containing the same information for redundancy purpose)
- RAM memory
- ROM memory
- OTP memory (Fuse)
- HMAC-SHA256, SHA256, and AES hardware engines
- Public Key cryptographic coprocessor
- A True Random Number Generator (TRNG)
- A Deterministic Random Bit Generator (DRBG) based on HMAC
- Environmental sensors

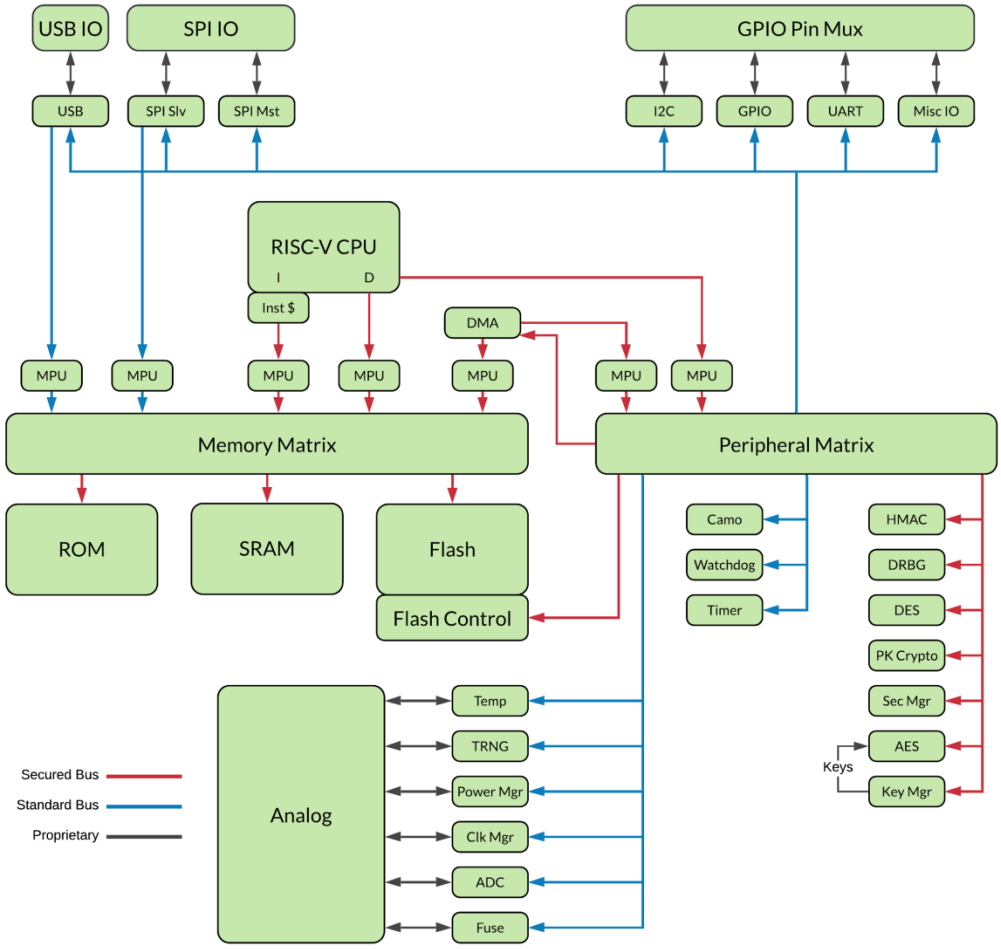


Figure 1 TOE overview

1.5.2.2 Interface

The TOE provides the following interfaces at its boundary:

- 3 SPI interfaces (2 masters, 1 slave)
- 6 I2C interfaces (3 masters, 3 slaves)
- 5 UART
- 1 USB interface
- PWM output

Available through 44 generic GPIO behind a multiplexer
None of these interfaces are used by the bootloader (except one SPI and one UART and one IO PIN) and crypto library.

1.5.3 Software

1.5.3.1 Bootloader

The TOE contains in its ROM code a bootloader that will be executed upon Power-on-Reset.
The Bootloader performs the initialization and configuration of the hardware including countermeasures and Random Number Generator. In addition, the Bootloader verifies the signature of the code loaded in internal flash before executing this code. The bootloader can also be told to erase the internal flash and write a new code.

1.5.3.2 Cryptographic Services

The TOE supports the following services implemented as a combination of hardware and software. All services are available through a software interface implemented by the Crypto Library.

RSA

- The RSA algorithm provides encryption with OAEP, PKCS#1 v1.5 and no padding
- The RSA algorithm provides decryption with no padding
- The RSA algorithm provides signature generation with PSS or PKCS#1 v1.5 padding and signature verification.
- The supported key size is: 2048 bits

ECC

- The ECC key generation algorithm computes a public key associated to a private key that can be used with ECDSA or ECDH
- The supported curve is: NIST P-256

ECDSA

- The ECDSA algorithm provides signature and verification functionality
- The supported curve is: NIST P-256

ECDH

- The ECDH algorithm provides key exchange functionality
- The supported curve is: NIST P-256

AES

- The AES algorithm provides encryption, decryption and MAC functionalities
- The following modes of operation are supported: CBC, ECB, CMAC, GCM and CTR
- The supported key sizes are 128, 192 and 256 bits
- The Crypto Library leverages the HW AES and provides different security configurations. Please refer to the user guidance documentation.

SHA

- The SHA-256 hardware engine is provided for various purposes including as a building block for HMAC
- The SHA-384 and SHA-512 are implemented in software

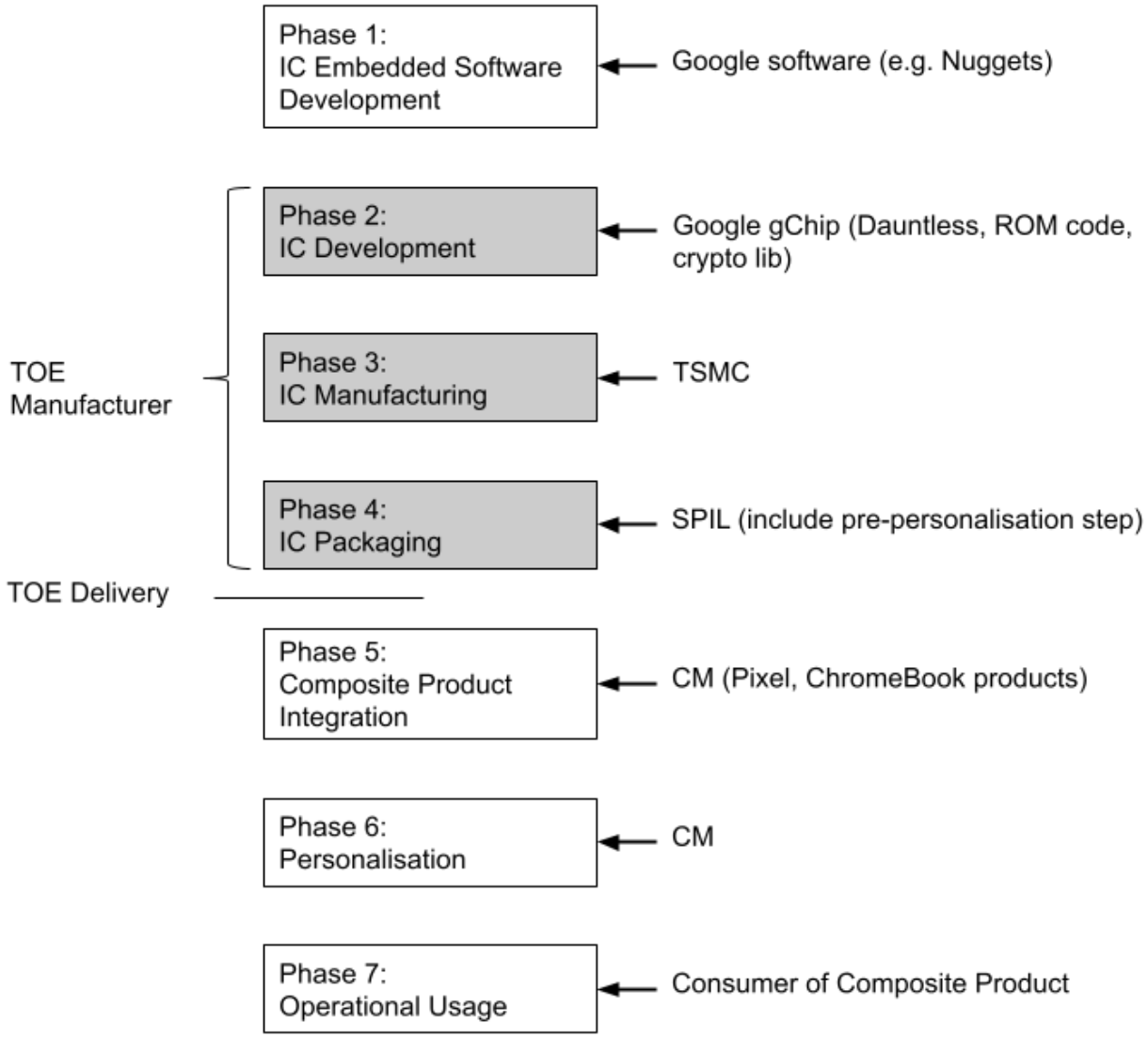
HMAC

- HMAC-SHA256/384/512 are provided to perform Keyed-hash functions.
- The supported key size is 64 to 512 bits[1]

[1] Keys of sufficient length shall be used. Please refer to relevant national or international standards for more details.

1.5.4 Life-Cycle

The TOE manufacturer is involved in steps 2 to 4 which are therefore subject to the Minimum Site Security Requirements.
The TOE is delivered in the form of Packaged IC pre-personalized to the Composite Product Integrator (Contract Manufacturer).



The following sites are involved in the TOE life-cycle:

Phase	Role	Entity	Address
3	Mask house	TSMC	Fab 2/5: 121, Park Ave. 3, Hsinchu Science Park, Hsinchu 300-77, Taiwan, R.O.C., and Fab8: 25, Li-Hsin Rd., Hsinchu Science Park, Hsinchu 300-78, Taiwan, R.O.C.
3	Wafer Fab & Sort	TSMC	Fab14A: 1-1, Nan-Ke North Rd., Tainan Science Park, Tainan 741-44, Taiwan, R.O.C. Fab 18: No.8 Beiyuan 2nd Rd., Tainan Science Park Tainan City 745-43, Taiwan, R.O.C.
3	IP-Merge	TSMC	Fab 3: 9, Creation Rd. 1, Hsinchu Science Park, Hsinchu 300-77, Taiwan, R.O.C.
4	Packaging	SPIL Changhua	No.8, Sec 2, Chang Hsin Rd., Hemei. Changhua, Taiwan 508, R.O.C.
4	Final Test	SPIL Hsinchu III	No. 1-1, R&D Rd. 2, Science-Based Industrial Park, Hsinchu, Taiwan 300, R.O.C.
4	SLT (pre-perso)	SPIL Hsinchu I	No.4, Creation Rd. 4, Science-Based Industrial Park, Hsinchu, Taiwan 300, R.O.C

More information on the sites involved in other life-cycle phases including development can be found in the Certification Report and documents referenced therein.

1.5.5 Required non-TOE hardware/software/firmware

As described in the protection profile that this Security Target claims conformance to (cf. Section 2.2), the Security IC Embedded Software is not part of the TOE and must be implemented by the user of the TOE. In order to use some of the interfaces described in Section 1.5.2.2, the user must implement the relevant functionality as part of this Security IC Embedded Software. More details are provided in the TOE user guidance documents.

1.6 TOE Description

1.6.1 Logical scope

The TOE offers the following logical security features:

- SF.Malfunction provides protection from malfunctions.
- SF.Test provides test functionality to be used during production.
- SF.Physical provides protection from physical attacks to the TOE and its memories.
- SF.Leak provides protection from side-channel leakage.
- SF.Crypto provides cryptographic functionality based on dedicated hardware engines as well as software.
- SF.MPU provides a memory protect unit that can be used to define access controls to memory.
- SF.Loader provides a bootloader that can be used to load IC Embedded Software, and that will ensure only authentic code is executed.

1.6.2 Physical scope

This section lists all the components that comprise the TOE and their identification.

Component	**Identification**	**Delivery form**	**Delivery method**
Bootloader stored in ROM	7f4bdb	(implicit within the chip	Secure shipment
IC hardware	Packaged chip H1D3M, H1D3P or H1D3C	chip	Secure shipment
Crypto Library	Cryptolib v1.3.10	Binary	Secure shipment
Product data sheet	H1D3M Datasheet v1.2 06/30/2021	PDF	Electronic download
Product data sheet	H1D3P Datasheet v1.2 06/30/2021	PDF	Electronic download
Product data sheet	H1D3C Datasheet v1.2 06/30/2021	PDF	Electronic download
Register Specification	H1D3 Register Specification 09/30/2021	PDF	Electronic download
Code Signing procedure	H1D3 Code Signing v1.1 09/28/2021	PDF	Electronic download
Flashing manual	H1D3 SPI flashing instructions v1.2	PDF	Electronic download

	10/8/2021		
CPU Reference Manual	Soteria Technical Reference Manual v1.3 10/8/2021	PDF	Electronic download
Preparatory Guidance	H1D3 Preparatory Guidance v3.2 7/6/2023	PDF	Electronic download
User Guidance	H1D3 User Guidance v1.3 09/28/2021	PDF	Electronic download
Crypto User Guidance	Cryptolib v1.3.10 API User Guidance 7/5/2023	PDF	Electronic download
Crypto User Guidance	Addendum Cryptolib v1.3.10 API User Guidance v1.1 9/11/2023	PDF	Electronic download

Software running on the TOE is only developed by Google internal customers which are given access to the guidance documentation through Google Drive on a need-to-know basis.
Contract manufacturer (CM) integrating the TOE have access to the datasheet through the TOE distributor (GUC) using sFTP.

2 Conformance Claim

2.1 CC conformance claim

H1D3 Secure Microcontroller with Crypto Library v1.3.10 and this Security Target claim conformance to version 3.1 Revision 5 of the Common Criteria for Information Technology Security Evaluation. This conformance is Part 2 extended and Part 3 conformant.

The following specification applies:

- Common Criteria for Information Technology Security Evaluation, Part 1 Introduction and general model, version 3.1, Revision 5, April 17
- Common Criteria for Information Technology Security Evaluation, Part 2 Security functional components, version 3.1, Revision 5, April 17
- Common Criteria for Information Technology Security Evaluation, Part 3 Security assurance components, version 3.1, Revision 5, April 17
- Common Methodology for Information Technology Security Evaluation, Version 3.1, Revision 5, April 2017

2.2 PP and package claims

This security target claims strict conformance to “Security IC Platform Protection Profile with Augmentation Packages”, Version 1.0 referenced BSI-PP-0084-2014.
The following packages defined in the Protection Profile are also claimed:

- Package AES
- Package Hash functions (SHA-256, SHA-384, SHA-512)

This Security Target claims the same augmented package as the Protection Profile:

- EAL4 augmented with ATE_DPT.2[2], ALC_DVS.2 and AVA_VAN.5.

2.3 Conformance Claim Rationale

The description of H1D3 Secure Microcontroller with Crypto Library v1.3.10 in Section 1 of this Security Target is consistent with the TOE definition in Section 1.2.2 of the Protection Profile. Specifically, the TOE consists of:

- Security IC (H1D3 Secure Microcontroller)
- IC Dedicated Software (Crypto Library v1.3.10)
- Guidance documentation describing usage of the TOE

[2] ATE_DPT.2 used to be part of EAL4 in an earlier version of the CC standard

This is consistent with paragraphs 9 and 10 in Section 1.2.2 of the Protection Profile.

In addition to the pre-defined packages from the Protection Profile, the following other cryptographic services are provided by the TOE:

- RSA
- ECDSA
- ECDH
- HMAC-256/384/512

Finally, the following other services are provided by the TOE:

- MPU
- Loader for authenticated data

The statement of security problem definition in this ST is consistent with the statement of security problem definition in the Protection Profile, because it consists of all standard threats, assumptions, and OSPs from the Protection Profile, as well as those defined in the augmentation packages that have been claimed. In order to address the additional services provided by the TOE, three additional OSPs P.Crypto-Service-Add, P.MPU, and P.Auth-Loader have been added in this ST. This does not conflict with the defined threats, assumptions, and OSPs from the Protection Profile, and is explicitly allowed as per Application Note 5 of the Protection Profile. The P.Auth-Loader policy is similar to the Loader Package 2 defined in the Protection Profile, except that the functionality provided by this TOE does not use a trusted channel but it ensures that only authentic data is accepted.

Furthermore, the statement of security objectives in this ST is consistent with the statement of security objectives in the Protection Profile, because it consists of all standard objectives for the TOE and for the operational environment from the Protection Profile, as well as those defined in the augmentation packages that have been claimed. Additional objectives for the TOE and environment have been added in order to enforce P.Crypto-Service-Add, P.MPU, and P.Auth-Loader, and these additional objectives do not mitigate any threats, enforce any OSPs, or address any assumptions from the Protection Profile as described in Section 4.3.

Finally, the statement of security requirements in this ST is consistent with the statement of security requirements in the Protection Profile, because it consists of all standard security functional requirements from the Protection Profile, as well as those defined in the augmentation packages that have been claimed. Additional SFRs have been added to meet the additional objectives defined in this Security Target. Two iterations are performed on the SFR component FCS_RNG.1 in this Security Target: FCS_RNG.1/TRNG and FCS_RNG.1/DRBG. The TRNG iteration matches the SFR FCS_RNG.1 from the Protection Profile by providing a *physical* random number generator. The additional requirement for a DRBG, which is not mandated by the Protection Profile, makes the security claims more strict as allowed in the context of strict conformance.

The SAR package claim in this Security Target is equal to the SAR package claim in the Protection Profile.

3 Security Problem Definition

3.1 Definition of assets

The assets protected by the TOE are

- the user data handled and stored in the Composite TOE
- the Security IC Embedded Software
- the security services provided by the TOE for the Security IC Embedded Software

The user data (including but not limited to the Security IC Embedded Software) are protected by the TOE security functionality in integrity while stored in Flash, SRAM, or OTP and confidentiality while stored in the Flash and SRAM areas. Additionally, the TOE security functionality ensures the correct operation of the provided security services.

The TOE security functionality is guaranteed by protecting the artifacts involved in the TOE life cycle.

The following artifacts are considered Restricted[3]:

- Design documentation
- Logical design data
- IC dedicated software
- Software development kit

The following artifacts are considered Critical[1] information and are protected as such:

- Configuration and pre-personalization data
- Test and calibration related data
- Samples with Debug enable

The following artifacts are considered Very Critical[1] information and are protected as such:

- Physical design data
- Photomasks

3.2 Threats

All threats defined in the Protection Profile including the claimed packages are applicable to this Security Target and are included here by reference.

Name	**Title**
T.Malfunction	Malfunction due to Environmental Stress

[3]As defined by Application of Attack Potential to Smartcards and Similar Devices, version 3.0

T.Abuse-Func	Abuse of Functionality
T.Phys-Probing	Physical probing
T.Physical-Manipulation	Physical manipulation
T.Leak-Inherent	Inherent Information Leakage
T.Leak-Forced	Forced Information leakage
T.RND	Deficiency of Random Numbers

3.3 Organizational Security policies

All OSPs defined in the Protection Profile including the claimed packages are applicable to this Security Target and are included here by reference.

Name	**Title**
P.Process-TOE	Identification during TOE Development and Production
P.Crypto-Service	Cryptographic services of the TOE

Additionally, the following OSPs are defined to include additional services not yet covered by augmentation packages of the Protection Profile.

Name	**Title**
P.Crypto-Service-Add	Additional cryptographic services of the TOE The TOE provides additional secure hardware based cryptographic services for the IC Embedded Software.
P.MPU	Memory Protection Unit The TOE provides an MPU that can be used to define memory access control based on regions in memory.
P.Auth-Loader	Loader for authenticated data The Loader functionality verifies the authenticity and version of the IC Embedded Software at boot time in order to ensure that only authentic IC Embedded Software are accepted, and that it is not possible to downgrade the IC Embedded Software by rolling back to an earlier but authentic version.

3.4 Security Assumptions

All assumptions defined in the Protection Profile including the claimed packages are applicable to this Security Target and are included here by reference.

Name	**Title**
A.Process-Sec-IC	Protection during Packaging, Finishing and Personalisation
A.Resp-Appl	Treatment of user data of the Composite TOE

4 Security Objectives

4.1 TOE security objectives

All TOE security objectives defined in the Protection Profile including the claimed packages are applicable to this Security Target and are included here by reference.

Name	**Title**
O.Malfunction	Protection against malfunctions
O.Abuse-Func	Protection against Abuse of Functionality
O.Phys-Probing	Protection against Physical Probing
O.Phys-Manipulation	Protection against Physical Manipulation
O.Leak-Inherent	Protection against Inherent Information Leakage
O.Leak-Forced	Protection against Forced Information Leakage
O.RND	Random Numbers
O.Identification	TOE Identification
O.AES	Cryptographic service AES
O.SHA	Cryptographic service Hash function

Additionally, the following security objectives are defined for the TOE, corresponding to services that are provided by the TOE:

Name	**Title and description**
O.CMAC	Cryptographic service CMAC The TOE provides secure hardware based cryptographic services implementing AES CMAC for computing Message Authentication Codes.
O.RSA	Cryptographic service RSA The TOE provides secure hardware based cryptographic services implementing RSA for encryption, decryption, signature generation, and signature verification.
O.ECC	Cryptographic service ECC The TOE provides secure hardware based cryptographic services implementing ECC key generation.

O.ECDSA	Cryptographic service ECDSA The TOE provides secure hardware based cryptographic services implementing ECDSA for signature generation, and signature verification.
O.ECDH	Cryptographic service ECDH The TOE provides secure hardware based cryptographic services implementing ECDH for key exchange.
O.HMAC	Cryptographic service HMAC The TOE provides secure hardware based cryptographic services implementing the keyed hash HMAC-SHA256 algorithm for computing Message Authentication Codes.
O.MPU	Memory Protection Unit The TSF enforces an access control policy on the memory based on user-configured memory regions.
O.Auth-Loader	Loader data authentication The TSF supports loading the IC Embedded Software, and authentication of the IC Embedded Software at boot time. It also performs self-tests of the boot code to ensure the integrity of this functionality.

4.2 Development and operational environment security

4.2.1 Security objectives for the Security IC Embedded Software

All security objectives for the Security IC Embedded Software defined in the Protection Profile are applicable to this Security Target and are included here by reference.

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

4.2.2 Security objectives for the operational environment

All security objectives for the operational environment defined in the Protection Profile are applicable to this Security Target and are included here by reference.

Name	**Title**
OE.Process-Sec-IC	Protection during composite product manufacturing

Additionally, the following objective for the environment is defined in this Security Target.

Name	Title
OE.Auth-Loading	Authenticity of the IC Embedded Software The authorized user must support the capability to provide the authenticity proof of the IC Embedded Software. Additionally, the authorized user must use the dedicated memory in order to prevent downgrade attacks on the software version.

4.3 Security objectives rationale

For the standard threats, assumptions, and OSPs from the Protection Profile, as well as those defined in the augmentation packages that have been claimed, the security objectives rationale is exactly as described in the Protection Profile and hence it is not repeated here.

For the OSPs defined in this Security Target, the following table gives an overview how the OSPs are addressed by the objectives:

Assumption, Threat, or OSP	Security Objectives
P.Crypto-Services-Add	O.CMAC O.RSA O.ECC O.ECDSA O.ECDH O.HMAC
P.MPU	O.MPU
P.Auth-Loader	O.Auth-Loader OE.Auth-Loading

The justification related to the OSP P.Crypto-Services-Add is as follows: The OSP is defined to provide additional secure hardware based cryptographic services for the IC Embedded Software, and each of the objectives O.CMAC, O.RSA, O.ECC, O.ECDSA, O.ECDH, and O.HMAC requires to provide a specific secure hardware based cryptographic service. Therefore, each objective contributes to addressing the OSP, and conversely the OSP is covered by these objectives.

The justification related to the OSP P.MPU is as follows: The OSP requires that a service is provided, and this service is directly implemented by the objective O.MPU.

The justification related to the OSP P.Auth-Loader is as follows: The OSP is directly implemented by the security objective for the TOE O.Auth-Loader and the security objective for the operational environment OE.Auth-Loading.

5 Extended Components Definition

5.1 From the underlying Protection Profile

The certified PP defines the following extended components, FCS_RNG.1, FMT_LIM.1, FMT_LIM.2, FAU_SAS.1 and FDP_SDC.1 and they are included here by reference.

Name	**Title**
FCS_RNG.1	Random Number Generation
FMT_LIM.1	Limited capabilities
FMT_LIM.2	Limited availability
FAU_SAS.1	Audit storage
FDP_SDC.1	Stored data confidentiality

5.2 Defined in this Security Target

No additional extended components are defined in this Security Target.

6 Security Requirements

All Security Functional Requirements (SFRs) of the TOE are presented in the following sections to support a better understanding of the combination of the PP and this Security Target. It is clearly stated which subset of SFRs is taken from the underlying protection profile or its functional packages and which are newly introduced.

Regarding the Security Assurance Requirements, the package claim is equal to that of the claimed Protection Profile as described in Section 2. As such, the SARs are included here by reference from the Protection Profile, including all performed operations.

6.1 Security Functional Requirements from the PP

6.1.1 Standard SFRs from the PP

The following table lists the standard SFRs from the Protection Profile that do not have any open operations. No refinements have been performed on these SFRs in this Security Target, and they are included by reference to the Protection Profile.

Name	**Title**
FRU_FLT.2	Limited Fault Tolerance
FPT_FLS.1	Failure with preservation of secure state
FMT_LIM.1	Limited capabilities
FMT_LIM.2	Limited availability
FPT_PHP.3	Resistance to Physical Attack
FDP_ITT.1	Basic internal transfer protection
FPT_ITT.1	Basic internal TSF data transfer protection
FDP_IFC.1	Subset information flow control

Additionally, the other standard SFRs from the Protection Profile have open operations. The following describes how these open operations are completed in this Security Target. Assignments and selections that were already completed in the PP are indicated with underlined text, whereas operations that have been performed in this ST are indicated as follows: assignments and selections are indicated by italicized text, iterations are performed by appending a forward slash (/) and a unique tag to the SFR, and refinements are indicated by strikethrough when text is removed or by bold-faced text when text is added.

The component FCS_RNG.1 from the Protection Profile has been iterated twice in this Security Target, i.e., to FCS_RNG.1/TRNG and FCS_RNG.1/DRBG. The iteration FCS_RNG.1/TRNG corresponds to

the SFR FCS_RNG.1 from the Protection Profile and finishes the operations that remain open in the Protection Profile. The iteration FCS_RNG.1/DRBG is based on the extended component and provides an additional deterministic random number generator as a service to the IC Security Embedded Software.

FAU_SAS.1 Audit storage

FAU_SAS.1.1 The TSF shall provide the test process before TOE Delivery with the capability to store

- *Initialization Data, Pre-personalisation Data, Public key for IC Embedded Software verification* in the *ROM area*.
- *Rollback counter, OTP data* in the *OTP area.*

FDP_SDC.1 Stored data confidentiality

FDP_SDC.1.1 The TSF shall ensure the confidentiality of the information of the user data while it is stored in the *Flash, and SRAM areas*.

FDP_SDI.2 Stored data integrity monitoring and action

FDP_SDI.2.1 The TSF shall monitor user data stored in containers controlled by the TSF for

- *ROM: persistent errors*
- *SRAM: single-bit errors per byte*
- *OTP: CRC-16 errors*
- *Flash: RSA-2048 signature*

on all objects, based on the following attributes: *ROM, SRAM, OTP, Flash*.

FDP_SDI.2.2 Upon detection of a data integrity error, the TSF shall

- *ROM, Flash: do not boot*
- *SRAM and OTP: raise an alarm*.

FCS_RNG.1/TRNG Random number generation – TRNG

FCS_RNG.1.1/TRNG 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 *continuously*. 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/TRNG The TSF shall provide *bits* 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.

FCS_RNG.1/DRBG Random number generation – DRBG

FCS_RNG.1.1/DRBG The TSF shall provide a *deterministic* random number generator that implements:

(DRG.3.1) If initialized with a random seed *using a PTRNG of class PTG.2 as random source*, the internal state of the DRBG shall *have at least 250 bits of entropy*.

(DRG.3.2) The DRBG provides forward secrecy.

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

FCS_RNG.1.2/DRBG The TSF shall provide random numbers that meet:

(DRG.3.4) The DRBG, initialized with a random seed *using a PTRNG of class PTG.2 as random source*, generates output for which 2^{35} strings of bit length 128 are mutually different with probability at least $1 – 2^{-18}$.

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

6.1.2 SFRs from claimed PP Packages

6.1.2.1 AES package

FCS_COP.1/AES Cryptographic operation – AES

FCS_COP.1.1/AES The TSF shall perform decryption and encryption in accordance with a specified cryptographic algorithm AES in *ECB mode, CBC mode,* ***GCM mode, CTR mode*** and cryptographic key sizes *128 bit, 192 bit, 256 bit* that meet the following: FIPS 197 [16]], NIST SP 800-38A [22], **NIST SP800-38D**

Application note: the references [16] and [22] are from the Protection Profile.

FCS_CKM.4/AES Cryptographic key destruction - AES

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

6.1.2.2 Hash functions package

FCS_COP.1/SHA Cryptographic operation – SHA

FCS_COP.1.1/SHA The TSF shall perform hashing in accordance with a specified cryptographic algorithm *SHA-256, SHA-384 and SHA-512* and cryptographic key sizes none that meet the following FIPS 180-4 [15]

Application note: the reference [15] is from the Protection Profile.

6.2 Security Functional Requirements added in this ST

6.2.1 Additional Cryptographic Services

FCS_COP.1/AES-CMAC Cryptographic operation – AES CMAC

FCS_COP.1.1/AES-CMAC The TSF shall ~~**perform**~~ ***compute*** *a message authentication code* in accordance with a specified cryptographic algorithm *AES CMAC* and cryptographic key sizes *128 bit, 192 bit, 256 bit* that meet the following: *NIST SP800-38B*.

FCS_COP.1/RSA Cryptographic operation – RSA

FCS_COP.1.1/RSA The TSF shall perform *encryption, signature generation and signature verification* in accordance with a specified cryptographic algorithm *RSA* and cryptographic key sizes *2048 bit* that meet the following: *PKCS #1, v2.2: RSAES-OAEP, RSAES-PKCS1-V1_5, RSAEP, RSADP, RSASSA-PSS, RSASSA-PKCS1-V1_5*.

FCS_COP.1.1/RSA The TSF shall perform *decryption* in accordance with a specified cryptographic algorithm *RSA* and cryptographic key sizes *2048 bit* that meet the following: *RSAEP, RSADP, RSASSA-PSS*.

FCS_CKM.4/RSA Cryptographic key destruction – RSA

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

FCS_COP.1/ECDSA Cryptographic operation – ECDSA

FCS_COP.1.1/ECDSA The TSF shall perform *signature generation and signature verification* in accordance with a specified cryptographic algorithm *ECDSA* and

cryptographic key sizes *256 bit* that meet the following: *FIPS PUB 186-4*.

Application note: Only NIST P-256 is supported for ECDSA

FCS_CKM.1/ECC Cryptographic key generation – ECC

FCS_CKM.1.1/ECC The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *ECC point multiplication* and specified cryptographic key sizes *256 bit* that meet the following: *FIPS PUB 186-4*.

FCS_CKM.4/ECC Cryptographic key destruction – ECC

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

FCS_COP.1/ECDH Cryptographic operation – ECDH

FCS_COP.1.1/ECDH The TSF shall perform *key exchange* in accordance with a specified cryptographic algorithm *ECDH* and cryptographic key sizes *256 bit* that meet the following: *NIST SP800-56A*.

FCS_COP.1/HMAC Cryptographic operation – HMAC

FCS_COP.1.1/HMAC The TSF shall perform *a message authentication code* in accordance with a specified cryptographic algorithm *HMAC-SHA256, HMAC-SHA384, HMAC-SHA512* and cryptographic key sizes *64 to 512 bits* that meet the following: *FIPS PUB198-1*.

FCS_CKM.4/HMAC Cryptographic key destruction – HMAC

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

6.2.2 MPU

FDP_ACC.1/MPU Subset Access Control – MPU

FDP_ACC.1.1/MPU The TSF shall enforce the *MPU SFP* (security function policy) on

- *Subjects: Software*
- *Objects: Memory areas*
- *Operations: Read/Write/Execute*

FDP_ACF.1/MPU Security attribute based access control – MPU

FDP_ACF.1.1/MPU The TSF shall enforce the *MPU SFP* to objects based on the following:

- *Software: permission level (USER level or MACHINE level)*
- *Memory areas:*

    - *Programmed regions*
        - *base address,*
        - *limit address,*
        - *access restrictions (which SFP operations are allowed depending on the software permission level)*
    - *Target address*

FDP_ACF.1.2/MPU The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:

- *If the currently executing software attempts to perform an operation on a memory target address, and this target address is contained in one of the programmed regions (i.e., the target address lies between the base address and the limit address of the region), the operation is allowed if and only if the region access restrictions allow it*.

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

FDP_ACF.1.4/MPU The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *Any access to a target address outside defined regions is denied*.

FMT_MSA.3/MPU Static attribute initialisation – MPU

FMT_MSA.3.1/MPU The TSF shall enforce the *MPU SFP* to provide *restrictive* default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2/MPU The TSF shall allow the *no one* to specify alternative initial values to override the default values when an object or information is created.

Application note: The text "when an object or information is created" means "upon boot" in the context of the MPU SFP.

FMT_MSA.1/MPU Management of security attributes – MPU

FMT_MSA.1.1/MPU The TSF shall enforce the *MPU SFP* to restrict the ability to *modify* the security attributes *region settings* to *MACHINE level software*.

FMT_SMR.1/MPU Security roles – MPU

FMT_SMR.1.1/MPU The TSF shall maintain the roles *USER level software, MACHINE level software*.

FMT_SMR.1.2/MPU The TSF shall be able to associate users with roles.

FMT_SMF.1/MPU Specification of Management Functions – MPU

FMT_SMF.1.1 The TSF shall be capable of performing the following management functions: *Modify the region settings*.

FIA_UID.2/MPU User identification before any action – MPU

FIA_UID.2.1/MPU The TSF shall require ~~each user~~ software controlled under the MPU SFP to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

6.2.3 Loader

FDP_ACC.1/Loader Subset Access Control – Loader

FDP_ACC.1.1/Loader The TSF shall enforce the *Loader SFP* on

- *Subjects: Bootloader*
- *Objects: IC Embedded Software*
- *Operations: load, execute*

FDP_ACF.1/Loader Security attribute based access control – Loader

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

- *Bootloader: public key*
- *IC Embedded Software: authentication state*

FDP_ACF.1.2/Loader The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:

- *When the bootloader loads IC Embedded Software, the operation is allowed*.
- *When the bootloader attempts to execute the IC Embedded Software (i.e., upon boot), the operation is allowed if and only if its authentication state is authenticated (by verification of the digital signature using the bootloader public key).*
- *In any other case, the operation is denied.*

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

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

FMT_MSA.3/Loader Static attribute initialization – Loader

FMT_MSA.3.1/Loader The TSF shall enforce the *Loader SFP* to provide *fixed* default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2/Loader The TSF shall allow the *no one* to specify alternative initial values to override the default values when an object or information is created.

Application note: the default values for the security attributes are fixed during the manufacturing of the TOE.

FMT_MSA.1/Loader Management of security attributes – Loader

FMT_MSA.1.1/Loader The TSF shall enforce the *Loader SFP* to restrict the ability to *modify* the security attributes *authentication state* to *the TSF*.

FMT_SMF.1/Loader Specification of Management Functions – Loader

FMT_SMF.1.1/Loader The TSF shall be capable of performing the following management functions: *Modify the authentication state of IC Embedded Software*.

FDP_ITC.1/Loader Import of user data without security attributes – Loader

FDP_ITC.1.1/Loader The TSF shall enforce the *Loader SFP* when importing user data, controlled under the SFP, from outside of the TOE.

FDP_ITC.1.2/Loader The TSF shall ignore any security attributes associated with the user data when imported from outside the TOE.

FDP_ITC.1.3/Loader The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE:
When importing user data controlled under the SFP, the TSF shall set the authentication state of the IC Embedded Software to unauthenticated. Upon next boot, the TSF shall

- *set the authentication state of the IC Embedded Software by verifying the associated digital signature on the imported user data controlled under the SFP,*
- *perform the access check on the execution of the IC Embedded Software according to the Loader SFP.*

Application note: The imported user data indicates which values from dedicated memory areas (which can include OTP and Flash) need to be included in the digital signature computation. This mechanism can be used by the IC Embedded Software developer to prevent rollback attacks and to bind versions of the IC Embedded Software to a dedicated IC through its serial number for testing purposes.

FPT_TDC.1/Loader Inter-TSF basic TSF data consistency – Loader

FPT_TDC.1.1 The TSF shall provide the capability to consistently interpret *digital signatures on IC Embedded Software* when shared between the TSF and another trusted IT product.

FPT_TDC.1.2 The TSF shall use *the interpretation rules corresponding to the digital signature scheme* when interpreting the TSF data from another trusted IT product.

FPT_TST.1/Loader TSF testing – Loader

FPT_TST.1.1/Loader The TSF shall run a suite of self tests *during initial start-up* to demonstrate the correct operation of *the following parts of the TSF:*

- *the random number generator required by FCS_RNG.1/TRNG.*

FPT_TST.1.2/Loader The TSF shall provide authorised users with the capability to verify the integrity of *the following parts of the TSF data:*

- *the contents of ROM,*
- *the contents of the OTP,*
- *the IC Embedded Software.*

FPT_TST.1.3/Loader The TSF shall provide authorised users with the capability to verify the integrity of *the following parts of the TSF:*

- *the ROM boot code,*
- *the public key used to verify the digital signature.*

Application note: Note that the IC Embedded Software is considered user data in this Security Target. However, this data is considered TSF data for the composite product. Additionally all tests are performed

during initial start-up, and hence “the capability” for authorized users to verify the integrity of TSF data or TSF corresponds to the capability of triggering the initial start-up.

6.3 Security Requirements Rationale

6.3.1 Security Functional Requirements

6.3.1.1 Mapping of requirements to objectives

For the standard objectives from the Protection Profile, as well as those defined in the augmentation packages that have been claimed, the security requirements rationale is exactly as described in the Protection Profile and hence it is not repeated here. The only addition is that the SFR FCS_RNG.1 from the Protection Profile has been iterated to FCS_RNG.1/TRNG and FCS_RNG.1/DRBG in this ST, and they consequently both help to meet the objective O.RNG, as they provide RNG services.

The following table gives an overview how the additional Security Functional Requirements defined in this Security Target are combined to meet the Security Objectives defined in this Security Target.

Security Objective	**Security Functional Requirements**
O.CMAC	FCS_COP.1/AES-CMAC FCS_CKM.4/AES
O.RSA	FCS_COP.1/RSA FCS_CKM.4/RSA
O.ECC	FCS_CKM.1/ECC FCS_CKM.4/ECC
O.ECDSA	FCS_COP.1/ECDSA FCS_CKM.4/ECC
O.ECDH	FCS_COP.1/ECDH FCS_CKM.4/ECC
O.HMAC	FCS_COP.1/HMAC FCS_CKM.4/HMAC

O.MPU	FDP_ACC.1/MPU FDP_ACF.1/MPU FMT_MSA.3/MPU FMT_MSA.1/MPU FMT_SMR.1/MPU FMT_SMF.1/MPU FIA_UID.2/MPU
O.Auth-Loader	FDP_ACC.1/Loader FDP_ACF.1/Loader FMT_MSA.3/Loader FMT_MSA.1/Loader FMT_SMF.1/Loader FDP_ITC.1/Loader FPT_TDC.1/Loader FPT_TST.1/Loader

The justification related to the security objective O.CMAC is as follows: FCS_COP.1/AES-CMAC requires the TOE to provide AES CMAC functionality, thereby directly meeting the objective. Additionally, FCS_CKM.4/AES describe how the AES keys are removed from the internal key storage of the TOE.

The justification related to the security objective O.RSA is as follows: FCS_COP.1/RSA requires the TOE to provide RSA encryption, decryption, signature generation and verification functionality, thereby directly meeting the objective. Additionally, FCS_CKM.4/RSA describes how the RSA key is removed from the internal key storage of the TOE.

The justification related to the security objective O.ECC is as follows: FCS_CKM.1/ECC requires the TOE to provide ECC key generation, thereby directly meeting the objective. Additionally, FCS_CKM.4/ECC describes how the generated key is removed from the internal key storage of the TOE.

The justification related to the security objective O.ECDSA is as follows: FCS_COP.1/ECDSA requires the TOE to provide ECDSA signature generation and verification, thereby directly meeting the objective. Additionally, FCS_CKM.4/ECC describes how the ECDSA key is removed from the internal key storage of the TOE.

The justification related to the security objective O.ECDH is as follows: FCS_COP.1/ECDH requires the TOE to provide ECDH key exchange functionality, thereby directly meeting the objective. Additionally, FCS_CKM.4/ECC describes how the ECDH key is removed from the internal key storage of the TOE.

The justification related to the security objective O.HMAC is as follows: FCS_COP.1/HMAC requires the TOE to provide HMAC functionality, thereby directly meeting the objective. Additionally, FCS_CKM.4/HMAC describes how the HMAC key is removed from the internal key storage of the TOE.

The justification related to the security objective O.MPU is as follows: FDP_ACC.1/MPU defines the subjects and objects involved in the MPU SFP, whereas FDP_ACF.1/MPU defines the security attributes

associated with these subjects and objects, as well as the access control rules that comprise the MPU SFP. FMT_MSA.3/MPU describes the initialization of the security attributes of the MPU SFP, whereas FMT_MSA.1/MPU describes how the authorized roles defined by FMT_SMR.1/MPU can manage the security attributes using the management functions provided by FMT_SMF.1/MPU. FIA_UID.2/MPU requires that the subjects covered by the MPU SFP are identified at all times. Together, these Security Functional Requirements describe the MPU SFP, which is an access control policy that requires the TOE to meet the security objective O.MPU.

The justification related to the security objective O.Auth-Loader is as follows: FDP_ACC.1/Loader defines the subjects and objects involved in the Loader SFP, whereas FDP_ACF.1/Loader defines the security attributes associated with these subjects and objects, as well as the access control rules that comprise the Loader SFP. FMT_MSA.3/Loader describes the initialization of the security attributes of the Loader SFP, whereas FMT_MSA.1/Loader describes how the TSF manages the security attributes using the management function defined by FMT_SMF.1/Loader. FDP_ITC.1/Loader describes how the Loader SFP is used to import user data from outside the TOE without security attributes, such that the TSF can assign the security attributes and apply the appropriate access controls according to the Loader SFP. This is supported by the requirement FPT_TDC.1/Loader, that requires the TOE to be capable of correctly interpreting the digital signature. The verification of this digital signature happens during initial start-up, together with the self-test on the integrity and correct functioning of the ROM boot code as required by FPT_TST.1/Loader. Together, these requirements help the TOE meet the objective of using the Loader functionality to load the IC Embedded Software.

6.3.1.2 Dependencies of the Security Functional Requirements

For the standard SFRs from the Protection Profile, as well as those defined in the augmentation packages that have been claimed, the security requirements rationale is exactly as described in the Protection Profile and hence it is not repeated here. Just as in the Protection Profile, the dependencies of FCS_COP.1/AES on FDP_ITC.1, FDP_ITC.2, or FCS_CKM.1 are not fulfilled, as it is left to the IC Embedded Software to choose how AES keys are provided to the AES functionality provided by this TOE. Additionally, just as in the Protection Profile, the dependencies of FCS_COP.1/SHA on FDP_ITC.1, FDP_ITC.2, or FCS_CKM.1, as well as on FCS_CKM.4 are not fulfilled, because no key is used.

For the SFRs defined in this Security Target, the following table shows how the SFR dependencies are met, or how they are justified if not.

SFR	**Dependencies**	**Met or justified**
FCS_COP.1/AES-CMAC	FDP_ITC.1, FDP_ITC.2, or FCS_CKM.1	Justified[4]
	FCS_CKM.4	FCS_CKM.4/AES

[4] For this functionality, it is left to the IC Embedded Software to choose how the keys are provided to the TOE. The TOE provides RNG functionality to support key generation.

FCS_COP.1/RSA	FDP_ITC.1, FDP_ITC.2, or FCS_CKM.1	Justified[4]
	FCS_CKM.4	FCS_CKM.4/RSA
FCS_CKM.4/RSA	FDP_ITC.1, FDP_ITC.2, or FCS_CKM.1	Justified[4]
FCS_COP.1/ECDSA	FDP_ITC.1, FDP_ITC.2, or FCS_CKM.1	FCS_CKM.1/ECC
	FCS_CKM.4	FCS_CKM.4/ECC
FCS_CKM.1/ECC	FCS_CKM.2, or FCS_COP.1	FCS_COP.1/ECDSA
	FCS_CKM.4	FCS_CKM.4/ECC
FCS_CKM.4/ECC	FDP_ITC.1, FDP_ITC.2, or FCS_CKM.1	FCS_CKM.1/ECC
FCS_COP.1/ECDH	FDP_ITC.1, FDP_ITC.2, or FCS_CKM.1	FCS_CKM.1/ECC
	FCS_CKM.4	FCS_CKM.4/ECC
FCS_COP.1/HMAC	FDP_ITC.1, FDP_ITC.2, or FCS_CKM.1	Justified[4]
	FCS_CKM.4	FCS_CKM.4/HMAC
FCS_CKM.4/HMAC	FDP_ITC.1, FDP_ITC.2, or FCS_CKM.1	Justified[4]
FDP_ACC.1/MPU	FDP_ACF.1	FDP_ACF.1/MPU
FDP_ACF.1/MPU	FDP_ACC.1	FDP_ACC.1/MPU
	FMT_MSA.3	FMT_MSA.3/MPU
FMT_MSA.3/MPU	FMT_MSA.1	FMT_MSA.1/MPU
	FMT_SMR.1	FMT_SMR.1/MPU
FMT_MSA.1/MPU	FDP_ACC.1 or FDP_IFC.1	FDP_ACC.1/MPU
	FMT_SMR.1	FMT_SMR.1/MPU

	FMT_SMF.1	FMT_SMF.1/MPU
FMT_SMR.1/MPU	FIA_UID.1	FIA_UID.2/MPU
FMT_SMF.1/MPU	None	N/a
FIA_UID.2/MPU	None	N/a
FDP_ACC.1/Loader	FDP_ACF.1	FDP_ACF.1/Loader
FDP_ACF.1/Loader	FDP_ACC.1	FDP_ACC.1/Loader
	FMT_MSA.3	FMT_MSA.3/Loader
FMT_MSA.3/Loader	FMT_MSA.1	FMT_MSA.1/Loader
	FMT_SMR.1	Justified[5]
FMT_MSA.1/Loader	FDP_ACC.1 or FDP_IFC.1	FDP_ACC.1/Loader
	FMT_SMR.1	Justified[6]
	FMT_SMF.1	FMT_SMF.1/Loader
FMT_SMF.1/Loader	None	N/a
FDP_ITC.1/Loader	FDP_ACC.1 or FDP_IFC.1	FDP_ACC.1/Loader
	FMT_MSA.1	FMT_MSA.1/Loader
FPT_TST.1/Loader	None	N/a
FPT_TDC.1/Loader	None	N/a

6.3.2 Security Assurance Requirements

The SARs were chosen according to the Protection Profile, and the rationale given by the Protection Profile for these requirements applies here as well.

[5] The management functions performed as part of the Loader SFP are only performed by the TSF. It is therefore not necessary to define authorized roles.

7 TOE Summary Specification

7.1 SF.Malfunction

SF.Malfunction implements several countermeasures against malfunctions. Using these countermeasures, the TOE meets FRU_FLT.2.

In case a malfunction is detected by the above countermeasures, the TOE will raise an alarm. This ensures that the TOE preserves a secure state and hence meets FPT_FLS.1.

7.2 SF.Test

SF.Test provides the test functionality that the TOE uses to write user data and as a result the TOE meets FAU_SAS.1.

Additionally, by disabling part of the test functionality during the manufacturing phase, and ensuring that the remaining test functionality cannot be used to compromise TOE assets, SF.Test ensures that the TOE meets FMT_LIM.1 and FMT_LIM.2.

7.3 SF.Physical

SF.Physical implements several countermeasures against physical attacks.
As a result, SF.Physical helps the TOE to meet FPT_PHP.3 and FPT_TST.1/Loader.

Additionally, SF.Physical implements countermeasures to help ensure the integrity of the data stored in memory.
As a result, SF.Physical ensures the TOE meets FDP_SDI.2.

Finally, SF.Physical provides countermeasures to help ensure the confidentiality of the data stored in memory.
As a result, SF.Physical ensures the TOE meets FDP_SDC.1.

7.4 SF.Leak

SF.Leak provides countermeasures against data leakage.
As a result, SF.Leak, ensures the TOE meets FDP_IFC.1, FDP_ITT.1, and FPT_ITT.1.

7.5 SF.RNG

SF.RNG provides two random number generators:

- TRNG, which ensures that the TOE meets FCS_RNG.1/TRNG, and
- DRBG, based on NIST SP800-90A HMAC SHA-256, which ensures that the TOE meets FCS_RNG.1/DRBG

7.6 SF.Crypto

SF.Crypto consists of a combination of hardware and software functionality. The following hardware engines are provided:

- AES (FCS_COP.1/AES, FCS_COP.1/AES-CMAC)
- SHA-256 (FCS_COP.1/SHA)
- HMAC SHA-256 (FCS_COP.1/HMAC)
- PKC co-processor (FCS_COP.1/RSA, FCS_COP.1/ECDSA, FCS_CKM.1/ECC, FCS_COP.1/ECDH)

The cryptographic support software provides the following functionality to meet the corresponding SFRs:

- AES (FCS_COP.1/AES, FCS_COP.1/AES-CMAC)
- SHA-256/384/512 (FCS_COP.1/SHA)
- RSA encryption, decryption, signature verification, signature generation (FCS_COP.1/RSA)
- ECDSA signature generation and signature verification (FCS_COP.1/ECDSA)
- ECC key generation (FCS_CKM.1/ECC)
- ECDH (FCS_COP.1/ECDH)
- HMAC SHA-256/384/512 (FCS_COP.1/HMAC)

SF.Crypto also ensures that the keys are securely cleared from the internal buffers after each cryptographic operation or key generation procedure. As a result, the TOE meets FCS_CKM.4/AES, FCS_CKM.4/RSA, FCS_CKM.4/ECC, and FCS_CKM.4/HMAC.

7.7 SF.MPU

SF.MPU implements an access control policy with the following properties:

1. At start-up, the MPU has a default configuration that rejects all accesses until the ROM code integrity has been verified and the CPU has been released (FMT_MSA.3/MPU)
2. It allows MACHINE level software to define memory regions with access restrictions determining whether read/write/execute operations are enabled (FMT_MSA.1/MPU, FMT_SMF.1/MPU, FMT_SMR.1/MPU)
3. When loading data from memory, storing data in memory, or fetching instructions from memory, the access control policy respectively checks whether read, write, or execute operations are allowed according to the defined memory regions and allows or denies the operation based on this check. (FDP_ACC.1/MPU, FDP_ACF.1/MPU)
4. All software is subject to this access control policy (FIA_UID.2/MPU)

As a result, the TOE meets all SFRs related to the MPU SFP.

7.8 SF.Loader

SF.Loader implements a Loader functionality which is governed by an access control policy with the following properties:

1. SF.Loader keeps the digital signature public key in OTP memory, which is initialised during manufacturing (FMT_MSA.3/Loader).
2. Any user can import IC Embedded Software from outside the TOE (FDP_ITC.1/Loader, FDP_ACF.1/Loader, FDP_ACC.1/Loader)
3. Upon every boot, SF.Loader will perform self-tests to ensure its own integrity, as well as verify the digital signature provided with the loaded IC Embedded Software to determine its authenticity (FDP_ACC.1/Loader, FDP_ACF.1/Loader, FDP_ITC.1/Loader, FMT_MSA.1/Loader, FPT_TDC.1/Loader, FPT_TST.1/Loader). The IC Embedded Software can indicate locations from dedicated memory (OTP and Flash) such that their values are included in the digital signature as part of this verification.

As a result, the TOE meets all SFRs related to the Loader SFP.

8 Bibliography

8.1 References to standards

[CC1]	Common Criteria for Information Technology Security Evaluation, Part 1 Introduction and general model, version 3.1, Revision 5, April 17
[CC2]	Common Criteria for Information Technology Security Evaluation, Part 2 Security functional components, version 3.1, Revision 5, April 17
[CC3]	Common Criteria for Information Technology Security Evaluation, Part 3 Security assurance components, version 3.1, Revision 5, April 17
[CEM]	Common Methodology for Information Technology Security Evaluation, Version 3.1, Revision 5, April 2017
[PP0084]	BSI-PP-0084-2014, Security IC Platform Protection Profile with Augmentation Packages, Version 1.0
[FIPS180-4]	FIPS PUB 180-4, Secure Hash Standard (SHS), August 2015
[FIPS186-4]	FIPS PUB 186-4, Digital Signature Standard (DSS), July 2013
[FIPS197]	FIPS PUB 197, Advanced Encryption Standard (AES), November 2001
[FIPS198-1]	FIPS PUB 198-1, The Keyed-Hash Message Authentication Code (HMAC), July 2008
[SP800-38A]	NIST SP 800-38A, Recommendation for Block Cipher Modes of Operation: Methods and Techniques, December 2001
[SP800-38B]	NIST SP800-38B, Recommendation for Block Cipher Modes of Operation: the CMAC Mode for Authentication, May 2005 (Updated 10/6/2016)
[SP800-38D]	NIST SP800-38D, Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC, November 2007
[SP800-56A]	NIST SP800-56A, Recommendation for Pair-Wise Key-Establishment Schemes Using Discrete Logarithm Cryptography, April 2018
[SP800-67]	NIST SP 800-67, Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher, November 2017
[PKCS#1]	PKCS #1, RSA Cryptography Standard, v2.2, October 2012

Legal Disclaimer

Disclaimers
Limited warranty and liability
Google does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of the information in the document. Google takes no responsibility for the content in this document if provided by an information source outside of Google. In no event shall Google be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Right to make changes
Google reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Authorized Use
Google products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of a Google product can reasonably be expected to result in personal injury, death or severe property or environmental damage. Google and its suppliers accept no liability for inclusion and/or use of Google products in such equipment or applications and therefore such inclusion and/or use is at your own risk.

Applications
Applications that are described herein for any of these products are for illustrative purposes only. Google makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification. You are responsible for the design and operation of your applications and products using Google products, and Google accepts no liability for any assistance with applications or your product design. It is your sole responsibility to determine whether the Google product is suitable and fit for your applications and products planned, as well as for the planned application and use of your customer(s). Google does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in your applications or products, or the application or use by your third-party customer(s).

Export control
This document as well as the item(s) described herein may be subject to export control regulations. Export might require prior authorization from competent authorities.

Trademarks Notice
All referenced brands, product names, service names and trademarks are the property of their respective owners.