IFX_CCI_00007Dh, IFX_CCI_00007Eh, IFX_CCI_00007Fh H11 with optional Crypto Suites

Security Target Lite

Release

About this document

Scope and purpose

This document is the Security Target for the Infineon IFX_CCI_00007Dh/7Eh/7Fh H11 security controllers.

Intended audience

Composite product developers, Common Criteria Evaluators and Certifiers.

Table of contents

Table of contents **2**
List of figures **4**
List of tables **4**
1 Introduction (ASE_INT) **6**
1.1 ST reference 6
1.2 TOE reference 6
1.3 TOE overview 6
1.3.1 TOE definition and usage 6
1.3.2 TOE major security features 6
1.4 TOE description 7
1.4.1 TOE components 7
1.4.1.1 TOE hardware 7
1.4.1.2 Firmware 9
1.4.1.3 Libraries 9
1.4.2 Physical scope 9
1.4.3 Logical scope 10
1.4.3.1 Non-TSF 11
1.4.4 TOE delivery 11
1.4.5 Production sites 12
1.4.6 Configurations 12
1.4.7 Initialisation with embedded software 12
2 Conformance (ASE_CCL) **14**
2.1 Conformance claims 14
2.1.1 PP claims 14
2.1.2 Package claims 14
2.2 Conformance rationale 14
3 Security Problem Definition (ASE_SPD) **15**
3.1 Threats 15
3.1.1 Threats from PP0084 15
3.1.2 Threats defined in this ST 15
3.1.3 Assets regarding the Threats 15
3.2 Organizational security policies 15
3.2.1 Organizational security policies from PP0084 15
3.2.2 Organizational security policies defined in this ST 15
3.3 Assumptions 16
3.3.1 Assumptions defined in [PP0084] 16
3.3.2 Assumptions defined in this ST 16
4 Security Objectives (ASE_OBJ) **17**
4.1 Security objectives for the TOE 17
4.1.1 Security objectives for the TOE defined in PP0084 17
4.1.2 Security objectives for the TOE defined in this ST 17
4.2 Security objectives for the operational environment (OE) 18
4.2.1 OEs defined in [PP0084] 18
4.2.2 OEs defined in this ST 18
4.3 Security objectives rationale 18
5 Extended Components Definition (ASE_ECD) **20**

5.1 Extended components defined in [PP0084] 20
5.2 Extended components defined in this ST 20
6 Security Requirements (ASE_REQ) 21
6.1 Security functional requirements 21
6.1.1 Hardware random number generators 21
6.1.2 Hardware based cryptographic services 22
6.1.3 TSF testing 23
6.1.4 Malfunctions 24
6.1.5 Abuse of Functionality 24
6.1.6 Physical Manipulation and Probing 24
6.1.7 Leakage 25
6.1.8 Application Firewall 25
6.1.8.1 Policy definition 26
6.1.8.2 SFRs 28
6.1.9 Authentication of the Security IC 31
6.1.10 Flash loader 31
6.1.10.1 SFRs added in this ST 33
6.1.11 Crypto Suite 34
6.1.11.1 AES 34
6.1.11.2 TDES 35
6.1.11.3 FFC 36
6.1.11.4 RSA 36
6.1.11.5 ECC 37
6.1.11.6 Hash 39
6.1.11.7 Random 39
6.2 Security assurance requirements 41
6.2.1 Security Policy Model (SPM) details 42
6.3 Security requirements rationale 43
6.3.1 Rationale for the Security Functional Requirements 43
6.3.1.1 Additional SFRs related to O.Firewall 43
6.3.1.2 Additional SFRs related to O.Ctrl_Auth_Loader 44
6.3.1.3 Additional SFRs related to O.Phys-Manipulation 44
6.3.1.4 Additional SFRs related to O.AES 44
6.3.1.5 Additional SFRs related to O.AES-CMAC 44
6.3.1.6 Additional SFRs related to O.TDES 44
6.3.1.7 Additional SFRs related to O.TDES-RMAC 45
6.3.1.8 Additional SFRs related to O.FFC 45
6.3.1.9 Additional SFRs related to O.RSA 45
6.3.1.10 Additional SFRs related to O.ECC 45
6.3.1.11 Additional SFRs related to O.Hash 45
6.3.1.12 Additional SFRs related to O.RND 45
6.3.2 Dependencies of Security Functional Requirements 45
6.3.3 Rationale of the Assurance Requirements 47
7 TOE Summary Specification (ASE_TSS) 48
7.1 SF_DPM: Device Phase Management 48
7.2 SF_PS: Protection against Snooping 49
7.3 SF_PMA: Protection against Modifying Attacks 49
7.4 SF_PLA: Protection against Logical Attacks 50
7.5 SF_HC: Hardware provided cryptography 50

7.6 SF_CS: Crypto Suite Services 51
8 Hash values of libraries 52
9 Cryptographic Table 54
Acronyms 56
References 57
Revision history 59

List of figures

Figure 1 TOE hardware 7

List of tables

Table 1 Hardware/Firmware components 9
Table 2 Libraries 9
Table 3 User guidance 10
Table 4 Forms of delivery 12
Table 5 TOE configuration options 12
Table 6 Order Options to initialize the TOE with customer software 13
Table 7 Threats from [PP0084] 15
Table 8 Organisational Security Policies from [PP0084] 15
Table 9 Memory region-based access control 16
Table 10 Assumptions from [PP0084] 16
Table 11 Security objectives for the TOE from [PP0084] 17
Table 12 Security Objectives for the TOE 17
Table 13 Security objectives for the operational environment from [PP0084] 18
Table 14 Additional Security objectives for the environment 18
Table 15 FCS_RNG.1/TRNG 21
Table 16 FCS_COP.1/AES 22
Table 17 FCS_CKM.4/AES 22
Table 18 FCS_COP.1/TDES 22
Table 19 FCS_CKM.4/TDES 23
Table 20 TSF testing 23
Table 21 FAU_SAS.1 24
Table 22 FDP_SDC.1 25
Table 23 FDP_SDI.2 25
Table 24 FDP_ACC.2/AF 28
Table 25 FDP_ACF.1/AF 29
Table 26 FMT_MSA.3/AF 29
Table 27 FMT_MSA.1/AF/S 29
Table 28 FMT_MSA.1/AF/NS 30
Table 29 FMT_SMF.1/AF 30
Table 30 FMT_SMR.1/AF 30
Table 31 FIA_API.1 31
Table 32 FMT_LIM.1/Loader 31
Table 33 FMT_LIM.2/Loader 31

Table 34 FTP_ITC.1 32
Table 35 FDP_ACC.1/Loader 32
Table 36 FDP_ACF.1/Loader 32
Table 37 FMT_MTD.1/Loader 33
Table 38 FMT_SMR.1/Loader 34
Table 39 FMT_SMF.1/Loader....34
Table 40 FIA_UID.2/Loader 34
Table 41 FCS_COP.1/CS/AES/<iter> 35
Table 42 Cryptographic table for FCS_COP.1/CS/AES/<iter> 35
Table 43 FCS_COP.1/CS/TDES/<iter> 35
Table 44 Cryptographic table for FCS_COP.1/CS/TDES/<iter>....36
Table 45 FCS_COP.1/CS/FFC/<iter>....36
Table 46 Cryptographic table for FCS_COP.1/CS/FFC/<iter> 36
Table 47 FCS_COP.1/CS/RSA/<iter> 36
Table 48 Cryptographic table for FCS_COP.1/CS/RSA/<iter> 37
Table 49 FCS_COP.1/CS/ECC/<iter> 37
Table 50 Cryptographic table for FCS_COP.1/CS/ECC/<iter>....38
Table 51 Certified elliptic curves 38
Table 52 FCS_CKM.1/CS/ECC....38
Table 53 FCS_COP.1/CS/Hash/<iter> 39
Table 54 Cryptographic table for FCS_COP.1/CS/Hash/<iter> 39
Table 55 FCS_RNG.1/CS/PTG2....39
Table 56 FCS_RNG.1/CS/PTG3....40
Table 57 SAR list and refinements 41
Table 58 SFRs excluded from SPM....43
Table 59 Rationale for SFRs related to O.Firewall....43
Table 60 Rationale for additional SFRs related to O.Ctrl_Auth_Loader 44
Table 61 Dependencies of SFRs....45
Table 62 TOE Security Features....48
Table 63 SHA256 hash values 52
Table 64 Cryptographic table 54

1 Introduction (ASE_INT)

1.1 ST reference

The ST has the title IFX_CCI_00007Dh, IFX_CCI_00007Eh, IFX_CCI_00007Fh H11 with optional Crypto Suites, Rev. 1.7 and is dated 2025-07-09.

1.2 TOE reference

The full TOE name is:

IFX_CCI_00007Dh, IFX_CCI_00007Eh, IFX_CCI_00007Fh, design step H11
with firmware version 80.506.04.1,
optional HSL version 04.05.0030,
optional UMSLC version 02.01.0040,
optional Crypto Suites version 4.06.002,
optional Crypto Suites version 4.08.001.

The TOE is identified by the components as described in the physical scope, chapter 1.4.2.

- The Hardware version, design step and Firmware version can be read out form the chip by the Generic Chip Identification Mode (GCIM). The procedure how to read that data is described in the Programmers Reference Manual.
- The correct library versions can be verified by the corresponding hash values as defined in chapter 8.

1.3 TOE overview

1.3.1 TOE definition and usage

The TOE consists of a smart card IC (Security Controller), firmware and user guidance meeting high requirements in terms of performance and security. The TOE is designed by Infineon Technologies AG and is intended to be used in smart cards for security-relevant applications and as developing platform for smart card operating systems according to the life cycle model from [PP0084]. The TOE is the platform for the Embedded Software but the Embedded Software itself is not part of the TOE. The TOE does not require any non-TOE hardware/software/firmware.

1.3.2 TOE major security features

- Dual CPU in lockstep mode to detect integrity errors during processing.
- Memory integrity protection
- Memory encryption
- Bus masking for security peripherals
- Hardware True RNG
- Symmetric coprocessor for AES and TDES encryption and decryption
- Coprocessor to accelerate long integer and modular arithmetic operations for asymmetric cryptography. It relies on the embedded software to implement securely cryptographic algorithms.
- Global alarm system with security life control
- Tearing safe NVM write
- Armv8-M compliant MPU and SAU

- Robust set of sensors and detectors
- Redundant alarm propagation and system deactivation principle
- Peripheral access control
- Leakage control of data dependent code execution
- Device phase management
- The optional Crypto Suites provides hardened cryptographic functions for AES, TDES, RSA, ECC, finite field DH, Hash functions and a physical and hybrid physical random number generator.
- Instruction Stream Signature (ISS) coprocessor. The ISS can optionally be used to protect the CPU instruction flow. The hardware-based integrity protection concept of the TOE already provides a very effective program flow protection, such that the ISS is not needed. The ISS can nevertheless be used for compatibility reasons or as a very conservative additional countermeasure.

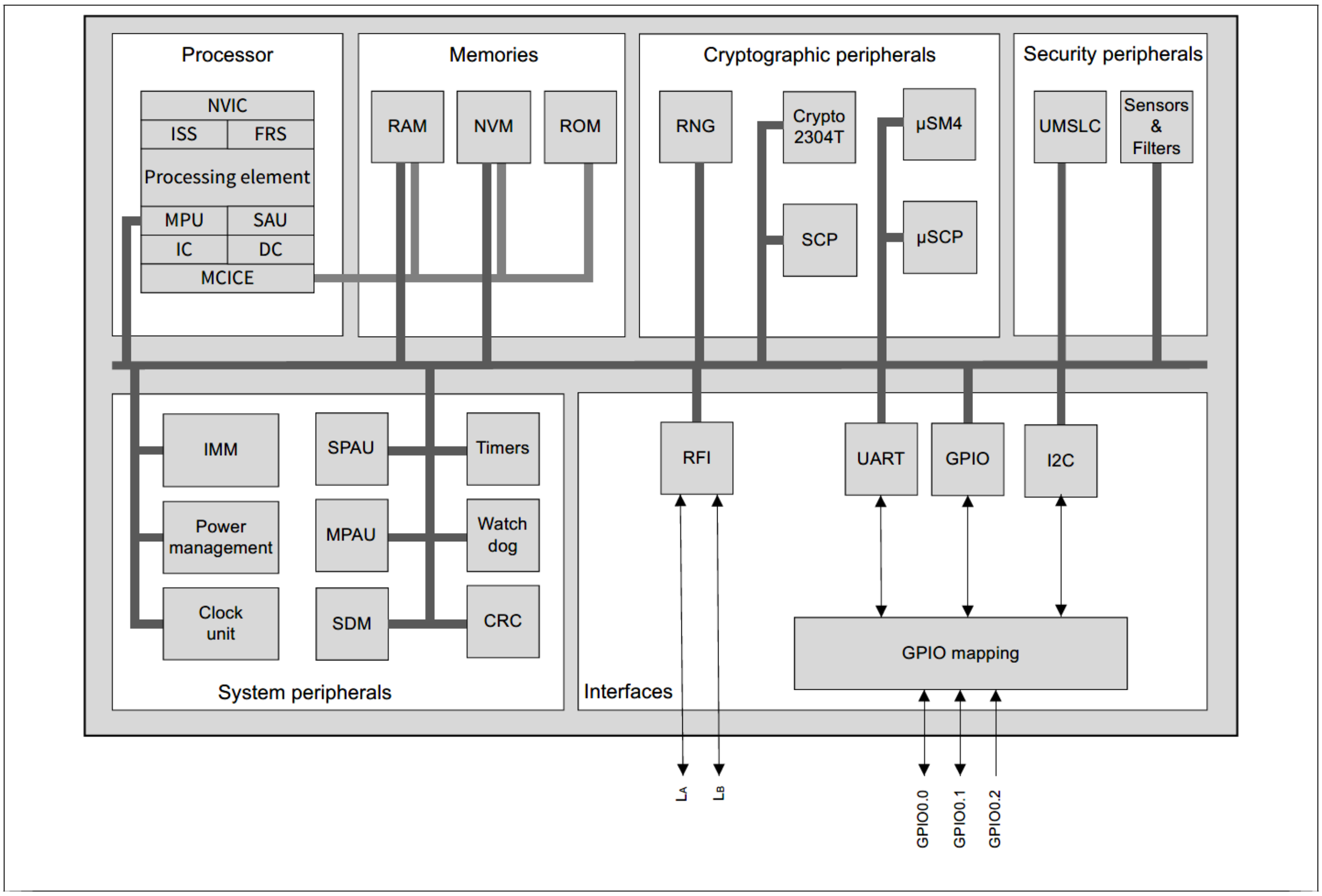
1.4 TOE description

1.4.1 TOE components

1.4.1.1 TOE hardware

Figure 1 shows schematically the TOE hardware.

Figure 1 TOE hardware



The TOE hardware consists of the following blocks:

- Processor
  - CPU according to Armv8-M mainline architecture.
  - Armv8-M compatible NVIC controller
  - Armv8-M compatible Memory Protection Unit (MPU) with 8 regions
  - Armv8-M compatible Security Attribution Unit (SAU) with 8 regions
  - Instruction Stream Signature (ISS) coprocessor
  - Fast Random Source (FRS) nonce generator coprocessor.
  - EDC protected caches for memory access and instruction fetch.
  - MCICE provides encryption and EDC protection for RAM, ROM and NVM
- Memories
  - encrypted and EDC protected ROM.
  - encrypted and EDC protected RAM.
  - encrypted and EDC protected NVM.
- Peripherals
  - Timers
  - Watchdog
  - CRC accelerator
- System peripherals
  - Clock unit
  - Interface Management Module (IMM)
  - Power Management
  - System Peripheral Access Unit (SPAU) to manage access to peripherals.
- Cryptographic peripherals
  - RNG according to class PTG.2 of [AIS 31]
  - Crypto2304T coprocessor for long modular integer arithmetic
  - SCP for secure AES and TDES computation
  - µSM4 accelerator for SM4 cryptographic algorithm.
  - µSCP accelerator for TDES cryptographic algorithm.
- Security peripherals
  - UMSLC
  - Sensors
- I/O Interfaces
  - UART for ISO 7816-3
  - I2C slave
  - RFI (ISO 14443/NFC contactless interface)
  - Miller interface
  - GPIO ports

The TOE has a global alarm system that puts the TOE into a secure state after tamper detection.

1.4.1.2 Firmware

The TOE Firmware consists of the Boot software, which provides secure start-up and contains the Flash Loader code.

1.4.1.3 Libraries

The TOE can be ordered with the following libraries to support the security coding of embedded software:

- UMSLC library to test the chips sensors.
- HSL library to provide tearing safe write for the NVM.
- Crypto Suite to provide security hardened cryptographic algorithms.

The TOE can be ordered with the NRG™ SW library. This is a proprietary cryptographic protocol for transport and ticketing applications. Please note that NRG™ is not part of the TSF.

The TOE can also be ordered with RF and CL libraries. Both libraries can be used to help the customer implementing ISO14443 contactless protocols. The NVM parts of the RF and CL libraries are not part of the TOE.

1.4.2 Physical scope

1.4.2.1 Hardware/Firmware

Table 1 Hardware/Firmware components

Component	Version
Hardware	IFX_CCI_00007Dh, IFX_CCI_00007Eh, IFX_CCI_00007Fh
Design step	H11
Firmware	80.506.04.1
Flash Loader	09.30.0001

Note: The TOE Firmware version includes the Boot software, which provides secure start-up, and the Flash Loader code. The Firmware version also contains the ROM parts of HSL, RF API and NRG API.

1.4.2.2 Libraries

The following libraries can be optionally ordered.

Table 2 Libraries

Component	Version
HSL	04.05.0030
UMSLC	02.01.0040
NRG™	06.10.0003 06.10.0005
Crypto Suite	4.06.002 4.08.001

Note: The user guidance for the UMSLC, NRG™ and HSL libraries is in the Programmers Reference Manual. The Crypto Suite library comes with dedicated user guidance.

1.4.2.3 User guidance documents

Table 3 User guidance

Component	Version	Date
TEGRION™ SLC26 (32-bit Security Controller – V19) Hardware Reference Manual	Revision 3.1	2023-08-08
SLx2 Security Controller Family Production and Personalization Manual Flash Loader V9	9.30	2023-08-11
SLC26 32-bit Security Controller - V19 Security Guidelines	1.00-3093	2025-06-04
TEGRION™ SLx2 security controller family Programmer's Reference Manual SLx2_DFP	1.8.0	2025-05-09
Crypto2304T V4, User Manual	3.0	2024-06-21
TEGRION™ SLC26 (32-bit Security Controller - V19) Errata sheet	6.0	2025-01-17
CS-SLC26V19 CryptoSuite 32-bit Security Controller User interface manual	4.06.002	2025-06-06
CS-SLC26V19 CryptoSuite 32-bit Security Controller User interface manual	4.08.001	2025-06-06

1.4.3 Logical scope

The logical scope of the TOE consists of the logical security features provided by the TOE. These features are listed in chapter 1.3.2. This chapter explains the features in more detail.

The following features of the TOE are part of the TSF:

- The Processor has a duplicated CPU running in lockstep mode to detect integrity errors. The CPU registers and the cache RAM are protected by 32-bit ECC codes used as an EDC.
- ROM, RAM and NVM content is cryptographically encrypted according to [AIS 46]
- ROM, RAM and NVM content is integrity protected by an EDC with at least 28 bits.
- A hardware true RNG according to class PTG.2 of [AIS 31].
- A symmetric coprocessor for performing masked AES and TDES ECB encryption.
- The data buses connecting the CPU and the cryptographic peripherals are encrypted using a bus encryption with dynamic keys which are changed in each transfer.
- Peripheral access control can be used to provide individual access control of all peripherals for the different security states of the processor (i.e., secure/non-secure, privilege/non-privilege).
- The chip has the following sensors.
  - voltage low and high
  - temperature low and high
  - low frequency
  - light fault attack detectors

  If the values are out of range a security alarm is issued.

- Security Life control is used to check proper working of sensors and alarm system by runtime triggered tests.
- In case the core or a peripheral detects a security violation it performs three countermeasures
  - goes into local alarm state.
  - propagates the alarm to the other peripherals and core which then go also into alarm state.
  - triggers a security reset.
- The HSL detects if NVM has not been correctly written due to a tearing event. The next time an HSL function is called, the embedded software is informed by the HSL that a tearing event has occurred. The HSL provides functions to correct the corrupted data by either roll-back or roll-forward.
- The Armv8-M Memory Protection Unit (MPU) and Security Attribution Unit (SAU) with 8 regions each are provided which can be used as a logical firewall for the embedded software.
- If the chip is switched to User mode it cannot be switched back to Test mode. If the Flash Loader is permanently disabled, it cannot be reactivated again.
- The optional Crypto Suites supporting.
  - RSA encryption and signature up to 4224 bits
  - ECDSA and ECDH up to 521 bits
  - Brainpool and NIST curves
  - AES 128, 192,256 in ECB and CBC chaining modes and CMAC
  - TDES 112, 168 bits in ECB and CBC chaining mode and Retail Mac
  - SHA-1 and SHA-2 Hash
  - RNG functions supporting PTG.2 and PTG.3
- The processor system comes with several supporting features to assist side-channel protected software implementations. One common software countermeasure is masking. However, the user needs to take special care when processing masked data together with its masks to avoid that they (or parts of them) are unintentionally combined in hardware resulting in a degradation of the desired side-channel security level. Therefore, the processor system has several measures in place to support the software in keeping the masked data and masks separated.
- Fast Random Source (FRS) nonce generator coprocessor. This RNG was not evaluated according to [AIS 31] and its output shall therefore not be used for applications requiring a certified RNG. It is used internally to support security features of the security architecture.
- Program flow integrity protection: The Instruction Stream Signature (ISS) coprocessor can optionally be used by the IC embedded software to detect illegal program flows and trigger an alarm.
- A coprocessor for accelerating long arithmetic operations to support RSA and ECC cryptography. This coprocessor has no dedicated security countermeasures. The embedded software must implement security countermeasures. Appropriate countermeasures are provided by the optional Crypto Suites library.

1.4.3.1 Non-TSF

The following features are part of the TOE, but not part of the TSF:

- NRG™ library

This ST does not make any security claims for the features listed in this section.

1.4.4 TOE delivery

The TOE delivery formats and delivery lifecycle according to [PP0084] application note 1 are shown in the following table.

Table 4 Forms of delivery

Component	Format	Life cycle	Delivery method
Hardware	bare die (sawn wafer)	3	Customer chooses delivery method. This ST describes under which circumstances transport protection is provided by the TOE.
	package	4	
Firmware	binary image	3 or 4[1]	In ROM/NVM of hardware
Libraries	object files	n/A	secure download via iShare
Documents	personalized PDF	n/A	secure download via iShare

1.4.5 Production sites

The TOE may be handled at different production sites, but the silicon is produced at TSMC fab 15 in Taiwan only. The production site can be determined by reading out the GCIM.

1.4.6 Configurations

This TOE is represented by various configurations called products. The module design, layout, and footprint, of all products are identical. The degree of freedom for configuring the TOE is predefined by Infineon Technologies AG. The following table shows TOE hardware/firmware configurations. The chip must be ordered with the desired NVM value. The value cannot be changed afterwards. Bill per Use is not supported.

Table 5 TOE configuration options

Component	Values	Identification
NVM	280, 360 kb	IFX mailbox
RAM	14, 20kb	IFX mailbox
uSCP	Available / not available	Hardware register
uSM4	Available / not available	Hardware register
RFI type A	Available / not available	Hardware register
RFI type B	Available / not available	Hardware register
RFI type F	Available / not available	Hardware register
RFI HBR	Available / not available	Hardware register
RFI VHBR	Available / not available	Hardware register
RFI AFM	Available / not available	Hardware register
NRG	Available / not available	Hardware register
RF antenna impedance	27, 56, 78 pf	GCIM

Further the Flash Loader can be configured in different ways as explained in the following section.

1.4.7 Initialisation with embedded software

This TOE is equipped with Flash Loader software (FL) to download user software, i.e., an operating system and applications. Various options can be chosen by the user to store software onto the NVM.

[1] depends on hardware delivery format

Table 6 Order Options to initialize the TOE with customer software

Option	TOE status
The user or/and a subcontractor downloads the software into the NVM. Infineon Technologies does not receive any user software.	The Flash Loader can be activated or reactivated by the user or subcontractor to download software into NVM. In case the Flash Loader is active, it may be either in life cycle stage “Pinletter” or “Activated”. When “Activated” a mutual authentication needs to be performed. In “Pinletter” a valid Pinletter provided by Infineon Technologies AG needs to be presented to enter “Activated” stage.
The user provides software to download into NVM to Infineon Technologies AG. The software is loaded into NVM during chip production.	There is no Flash Loader present.
The user provides software to download into NVM to Infineon Technologies AG. The software is loaded into NVM during chip production.	The Flash Loader is blocked by Infineon but can be activated or reactivated by the user or subcontractor to download software into NVM. The user is required to provide a reactivation procedure as part of the software to Infineon Technologies AG.
The user provides software to download into NVM to Infineon Technologies AG. The software is loaded into NVM during chip production.	The Flash Loader is active. The user can either download software or activate the software already present in NVM.

2 Conformance (ASE_CCL)

2.1 Conformance claims

This ST and TOE claim conformance to

- [CC2] extended
- [CC3] conformant

2.1.1 PP claims

This ST is strictly conformant to [PP0084]. The assurance level is EAL6 with the augmentation ALC_FLR.1.

The Security IC Platform Protection Profile with Augmentation Packages is registered and certified by the Bundesamt für Sicherheit in der Informationstechnik (BSI) under the reference [PP0084].

2.1.2 Package claims

This ST claims conformance to the following additional packages taken from [PP0084]:

- Package Authentication of the Security IC, section 7.2, conformant. This package is only claimed in case the Flash Loader is not permanently deactivated.
- Package Loader, Package 1: Loader dedicated for usage in secured environment only, section 7.3.1, conformant.
- Package Loader, Package 2: Loader dedicated for usage by authorized users only, section 7.3.2, augmented.
- Package TDES: section 7.4.1, augmented.
- Package AES: section 7.4.2, augmented.

The assurance level for the TOE is EAL6 augmented with the component ALC_FLR.1. Therefore, this ST is package-augmented to the packages in [PP0084].

2.2 Conformance rationale

The TOE is a typical security IC as defined in [PP0084].

The security problem definition of [PP0084] is enhanced by adding the Organisational Security Policy P.Firewall due to addition of the Armv8-M Memory Protection Unit and Security Extension. The security target remains conformant to [CC1] due to claim 289 as the possibility to introduce additional restrictions is given. The security target fulfils the strict conformance claim of [PP0084] due to application note 5.

3 Security Problem Definition (ASE_SPD)

3.1 Threats

3.1.1 Threats from PP0084

The following threats are defined and described in [PP0084] sections 3.2 and 7.2.1.

Table 7 Threats from [PP0084]

Threat	Description
T.Phys-Manipulation	Physical Manipulation
T.Phys-Probing	Physical Probing
T.Malfunction	Malfunction due to Environmental Stress
T.Leak-Inherent	Inherent Information Leakage
T.Leak-Forced	Forced Information Leakage
T.Abuse-Func	Abuse of Functionality
T.RND	Deficiency of Random Numbers
T.Masquerade_TOE	Masquerade the TOE

3.1.2 Threats defined in this ST

There are no additional threats defined in this ST.

3.1.3 Assets regarding the Threats

The asset description from [PP0084] section 3.1 applies.

3.2 Organizational security policies

3.2.1 Organizational security policies from PP0084

The organizational policies from [PP0084] sections 3.3, 7.3.1, 7.3.2 and 7.4 are applicable.

Table 8 Organisational Security Policies from [PP0084]

OSP	Description
P.Process-TOE	Protection during TOE Development and Production
P.Crypto-Service	Cryptographic services of the TOE
P.Lim_Block_Loader	Limiting and Blocking the Loader Functionality
P.Ctrl_Loader	Controlled usage to Loader Functionality

3.2.2 Organizational security policies defined in this ST

This ST defines an additional organisational security policy specific to the MPU Extension and Security Extension.

Table 9 Memory region-based access control

OSP	Definition
P.Firewall	The TOE must enable the IC dedicated software and the end-user embedded software to manage and control access to regions in memory.

3.3 Assumptions

3.3.1 Assumptions defined in [PP0084]

The TOE assumptions about the operational environment are defined and described in [PP0084] section 3.4.

Table 10 Assumptions from [PP0084]

Assumption	Description
A.Process-Sec-IC	Protection during Packaging, Finishing and Personalization
A.Resp-Appl	Treatment of User Data

3.3.2 Assumptions defined in this ST

There are no additional assumptions defined in this ST.

4 Security Objectives (ASE_OBJ)

4.1 Security objectives for the TOE

4.1.1 Security objectives for the TOE defined in PP0084

Table 11 Security objectives for the TOE from [PP0084]

Objective	Description
O.Phys-Manipulation	Protection against Physical Manipulation
O.Phys-Probing	Protection against Physical Probing
O.Malfunction	Protection against Malfunctions
O.Leak-Inherent	Protection against Inherent Information Leakage
O.Leak-Forced	Protection against Forced Information Leakage
O.Abuse-Func	Protection against Abuse of Functionality
O.Identification	TOE Identification
O.RND	Random Numbers
O.Cap_Avail_Loader	Capability and availability of the Loader
O.Ctrl_Auth_Loader	Access control and authenticity for the Loader
O.Authentication	Authentication to external entities
O.AES	Cryptographic service AES
O.TDES	Cryptographic service DES

4.1.2 Security objectives for the TOE defined in this ST

Table 12 Security Objectives for the TOE

Objective	Definition
O.Firewall	**Firewall based Access Control** The TOE must provide the IC dedicated software and the end-user embedded software with the capability to define restricted memory access and code execution to memory addresses. The TOE must enforce the access of software to these memory regions depending on access attributes.
O.FFC	**Finite Field cryptographic services** The TOE shall provide the following specific security functionality to the Smartcard Embedded Software: Finite Field cryptographic services
O.RSA	**RSA cryptographic services** The TOE shall provide the following specific security functionality to the Smartcard Embedded Software: Rivest-Shamir-Adleman Cryptography (RSA)
O.ECC	**Elliptic Curve cryptographic services** The TOE shall provide the following specific security functionality to the Smartcard Embedded Software: Elliptic Curve Cryptography (ECC)
O.AES-CMAC	**AES Cryptographic services** The TOE provides secure cryptographic services for AES CMAC generation.

Objective	Definition
O.TDES-RMAC	**TDES Cryptographic services** The TOE provides secure cryptographic services for TDES Retail MAC generation.
O.Hash	**Cryptographic service hash** The TOE provides secure cryptographic services for SHA-1 and SHA-2 generation.

4.2 Security objectives for the operational environment (OE)

4.2.1 OEs defined in [PP0084]

Table 13 Security objectives for the operational environment from [PP0084]

Objective	Description
OE.Resp-Appl	Treatment of user data of the Composite TOE
OE.Process-Sec-IC	Protection during composite product manufacturing
OE.Lim_Block_Loader	Limitation of capability and blocking the Loader
OE.Loader_Usage	Secure communication and usage of the Loader
OE.TOE_Auth	External entities authenticating of the TOE

Note: OE.TOE_Auth is available if the Flash Loader is available.

4.2.2 OEs defined in this ST

Table 14 Additional Security objectives for the environment

Environmental objective	Description
OE.Secure_Delivery (only applicable if Flash Loader deactivated (i.e. cases 2 and 3 from Table 6)	When the TOE is ordered with a disabled Flash Loader, it does not provide transport protection. Therefore, technical and / or organisational security procedures (e.g. a custom mutual authentication mechanism or a security transport) should be put in place by the customer to secure the personalized TOE during delivery as required by the security needs of the loaded IC Embedded Software.

4.3 Security objectives rationale

The security objectives rationale of the TOE is defined and described in [PP0084] section 4.4, 7.3.1, 7.3.2 and section 7.4.2.

The objectives O.Firewall added in this ST cover the organisational security policy P.Firewall that states that IC dedicated software and end-user embedded software must be able to manage and control access to regions in memory.

The objectives O.FFC, O.RSA, O.ECC, O.AES-CMAC, O.TDES-RMAC and O.Hash cover the policy P.Crypto-Service. This policy intends to allow adding various cryptographic services to the TSF.

The objective OE.Secure_Delivery requires the customers to provide transport protection in case the TOE is delivered with flash loader deactivated (i.e. cases 2, 3 from Table 6). In that case O.Authentication is not available and needs to be compensated by customer measures. This environmental objective counters T.Masquerade_TOE.

5 Extended Components Definition (ASE_ECD)

5.1 Extended components defined in [PP0084]

The [PP0084] defines the following extended components used in this ST:

- FMT_LIM.1
- FMT_LIM.2
- FAU_SAS.1
- FDP_SDC.1
- FCS_RNG.1
- FIA_API.1

5.2 Extended components defined in this ST

There are no extended components defined in this ST.

6 Security Requirements (ASE_REQ)

6.1 Security functional requirements

For the CC operations the following convention is used:

- CC operations which have been already completed in [PP0084] or [AIS 31] are typeset without underline.
- CC (nested) iteration operations are started by a slash "/" symbol, followed by an iteration identifier text. Iterations may be recursively nested.
- CC operations which are completed in this ST are underlined and the assigned footnote shows the original template text. Iteration operations are typed in normal font (i.e., without underline).

6.1.1 Hardware random number generators

Random numbers generation according to **Class PTG.2** of [AIS 31].

Table 15 FCS_RNG.1/TRNG

FCS_RNG.1/TRNG	**Random Number Generation**
Hierarchical to	No other components.
Dependencies	No dependencies.
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[1]. (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[2]. 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 32-bit numbers[3] that meet

[1] [selection: prevents the output of any internal random number that depends on some raw random numbers that have been generated after the total failure of the entropy source, generates the internal random numbers with a post-processing algorithm of class DRG.2 as long as its internal state entropy guarantees the claimed output entropy]

[2] [selection: externally, at regular intervals, continuously, applied upon specified internal events].

[3] [selection: bits, octets of bits, numbers [assignment: format of the numbers]]

FCS_RNG.1/TRNG	**Random Number Generation**
	(PTG.2.6) Test procedure A (None)[1] 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.

6.1.2 Hardware based cryptographic services

Table 16 FCS_COP.1/AES

FCS_COP.1/AES	**Cryptographic operation**
Hierarchical to	No other components.
Dependencies	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4
FCS_COP.1.1/AES	The TSF shall perform decryption and encryption in accordance with a specified cryptographic algorithm AES in ECB mode[2] and cryptographic key sizes 128 bit, 192 bit, 256 bit[3] that meet the following: [FIPS 197], [SP 800-38A].

Note: The input to the AES algorithm must be provided in two XOR shares. By fixing one share to zero a standard ECB mode results.

Table 17 FCS_CKM.4/AES

FCS_CKM.4/AES	**Cryptographic key destruction**
Hierarchical to	No other components.
Dependencies	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]
FCS_CKM.4.1/AES	The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method overwriting or zeroing[4] that meets the following: None[5].

Table 18 FCS_COP.1/TDES

FCS_COP.1/TDES	**Cryptographic operation**
Hierarchical to	No other components.
Dependencies	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4
FCS_COP.1.1/TDES	The TSF shall perform decryption and encryption in accordance with a specified cryptographic algorithm TDES in ECB mode[6] and

[1] [assignment: additional standard test suites]
[2] [selection: 128 bit, 192 bit, 256 bit]
[3] [selection: 128 bit, 192 bit, 256 bit]
[4] [assignment: cryptographic key destruction method]
[5] [assignment: list of standards]
[6] [selection: 128 bit, 192 bit, 256 bit]

FCS_COP.1/TDES	Cryptographic operation
	cryptographic key sizes 112bit, 168bit[1] that meet the following: [SP 800-67], [SP 800-38A].

Note: The input to the TDES algorithm must be provided in two XOR shares. By fixing one share to zero standard ECB mode results.

Table 19 FCS_CKM.4/TDES

FCS_CKM.4/TDES	Cryptographic key destruction
Hierarchical to	No other components.
Dependencies	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]
FCS_CKM.4.1/TDES	The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method overwriting or zeroing[2] that meets the following: None[3].

6.1.3 TSF testing

An attacker may try to circumvent the alarm system and secure wiring by physical manipulation (e.g. by cutting alarm lines). To counter those threats, the chip provides the User Mode Life Cycle (UMSLC) tests to check the integrity of those security features. Those test functions are provided as a software library and can be triggered on demand of the Embedded Software of the Composite TOE.

Table 20 TSF testing

FPT_TST.1	TSF testing
Hierarchical to	No other components.
Dependencies	No dependencies.
FPT_TST.1.1	The TSF shall run a suite of self-tests at the request of the authorized user[4] to demonstrate the correct operation of alarm test and security optimized wiring tests[5].
FPT_TST.1.2	The TSF shall provide authorized users with the capability to verify the integrity of the boot code[6].
FPT_TST.1.3	The TSF shall provide authorized users with the capability to verify the integrity of alarm behaviour and security optimized wiring[7].

Note: If the integrity of the boot code is violated, a security reset is triggered. The authorized user (i.e. the embedded software) can check if a security reset has been performed by reading the reset status register.

[1] [selection: 128 bit, 192 bit, 256 bit]
[2] [assignment: cryptographic key destruction method]
[3] [assignment: list of standards]
[4] [selection: during initial start-up, periodically during normal operation, at the request of the authorized user, at the conditions [assignment: conditions under which self-test should occur]]
[5] [selection: [assignment: parts of TSF], the TSF]
[6] [selection: [assignment: parts of TSF data], TSF data]
[7] [selection: [assignment: parts of TSF], TSF]

6.1.4 Malfunctions

This chapter relates to the section "Malfunctions" in [PP0084] ch. 6.1.

The SFRs FRU_FLT.2 and FPT_FLS.1 are specified in [PP0084].

Secure state of the TOE

Application note 14 of FPT_FLS.1 requires to define the secure state of the TOE.

Definition: A **secure state** of the TOE is either a correct operation or one of the following exceptional states

- security reset
- global deactivation of the TOE (a.k.a. alarm state)
- fault handler

6.1.5 Abuse of Functionality

This chapter relates to the section "Abuse of Functionality" in [PP0084] ch. 6.1.

The SFRs FMT_LIM.1 and FMT_LIM.2 are specified in [PP0084].

Table 21 FAU_SAS.1

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[1] with the capability to store the initialization data and/or pre-personalization data and/or supplements of the security IC embedded software[2] in the access protected and not changeable areas of the non-volatile memory[3].

6.1.6 Physical Manipulation and Probing

This chapter relates to the section "Physical Manipulation and Probing" in [PP0084] ch. 6.1.

The SFR FPT_PHP.3 is specified in [PP0084].

Automatic response of the TOE

Application note 19 of FPT_PHP.3 requires to define the automatic response of the TOE.
Definition: An **automatic response of the TOE** means entering a secure state of the TOE.

[1] [assignment: list of subjects]
[2] [assignment: list of audit information]
[3] [assignment: type of persistent memory]

Note on FPT_PHP.3 from [PP0084]

The methods as follows claim resistance towards leakage only in the context of the PACE protocols as defined in [ICAO]. Please note that obligations from the user guidelines with respect to these methods must be respected by the IC embedded software. The composite evaluator must assess the implementation of the PACE protocols.

For details, see confidential Security Target.

Table 22 FDP_SDC.1

FDP_SDC.1	**Stored data confidentiality**
Hierarchical to	No other components.
Dependencies	No dependencies
FDP_SDC.1.1	The TSF shall ensure the confidentiality of the information of the user data while it is stored in the RAM, ROM, and NVM[1].

Table 23 FDP_SDI.2

FDP_SDI.2	**Stored data integrity monitoring and action**
Hierarchical to	FDP_SDI.1
Dependencies	No dependencies.
FDP_SDI.2.1	The TSF shall monitor user data stored in containers controlled by the TSF for EDC integrity errors[2] on all objects, based on the following attributes: the corresponding EDC value with a length of at least 28 bits in the RAM, ROM, and NVM[3].
FDP_SDI.2.2	Upon detection of a data integrity error, the TSF shall enter a secure state[4].

6.1.7 Leakage

This chapter relates to the section “Leakage” in [PP0084] ch. 6.1.

The SFRs FDP_ITT.1, FPT_ITT.1 and FDP_IFC.1 are specified in [PP0084].

6.1.8 Application Firewall

The Application Firewall allows the embedded software to execute in four security levels and to assign access conditions to the address space related to the security levels. The security levels are:

- secure privilege
- secure non-privilege
- non-secure privilege
- non-secure non-privilege

[1] [assignment: memory area]
[2] [assignment: integrity errors]
[3] [assignment: user data attributes]
[4] [assignment: action to be taken]

The policy allows the embedded software to enforce the following trust relationship

- secure doesn't trust non-secure independent of the privilege level.
- secure privilege doesn't trust secure non-privilege.
- non-secure privilege doesn't trust non-secure non-privilege.

6.1.8.1 Policy definition

Subjects:

- Processor

Objects:

- Memory addresses

Operations:

- FETCH(x): any instruction fetch from address x.
- READ(x): any read access from address x.
- WRITE(x): any write access to address x.
- SG: secure gateway instruction.
- BNS: any of the branch to non-secure code instructions.
- FNC_RETURN: return from secure mode to non-secure mode.
- HANDLER_S: call of any secure handler code. Of specific importance for this policy are the following handlers:
  - MEMFAULT_S: memory fault handler in secure mode
  - SECFAULT: security fault handler
- HANDLER_NS: call of any non-secure handler code. Of specific importance for this policy is the following handler:
  - MEMFAULT_NS: memory fault handler in non-secure mode
- EXC_RETURN: return from handler code.

Security attributes for processor:

- sec: Boolean attribute designating secure/non-secure with values.
  - true: processor is in secure mode.
  - false: processor is non-secure mode.

- handlermode: Boolean attribute designating handler / thread mode:
  - true: processor is in handler mode.
  - false: processor is in thread mode.

- nPriv_S: Boolean attribute designating privilege mode when sec = true with values.
  - true: processor runs in non-privilege mode.

  - false: processor runs in privilege mode.

- nPriv_NS: Boolean attribute designating privilege mode when sec = false with values.
  - true: processor runs in non-privilege mode.
  - false: processor runs in privilege mode.

Security attributes for addresses:

- PO(x): Boolean attribute assigned to address x.
  - true: Only privilege mode has access.
  - false: Privilege and non-privilege mode have access.

- acc(x): {N, R, RW} attribute assigned to address x.
  - N: no access allowed.
  - R: write access is declined
  - RW: read and write access is not declined.

- sec(x): {S, NS, NSC} attribute assigned to address x.
  - S: secure address.
  - NS: non-secure address.
  - NSC: secure address which is callable from non-secure address.

- XN(x): Boolean variable assigned to address x.
  - true: instruction fetch is declined.
  - false: instruction fetch is not declined.

Definitions:

- privileged := handlermode or (not nPriv_S and sec) or (not nPriv_NS and not sec)

Rules:

1. FETCH(x) is declined if
   (sec = false and sec(x) = S)
   or (sec = false and sec(x) = NSC and FETCH(x) ≠ SG)

2. READ(x) is declined if
   sec = false and sec(x) ≠ NS
3. WRITE(x) is declined if
   sec = false and sec(x) ≠ NS

4. FETCH(x) is declined if
   (privileged = false and PO(x) = true)
   or (XN(x) = true)
   or (acc(x) = N)

5. READ(x) is declined if
   (privileged = false and PO(x) = true)
   or (acc(x) =N)
6. WRITE(x) is declined if
   (privileged = false and PO(x) = true)
   or (acc(x) ≠ RW)

7. If one of rules 1, 2, 3 apply then the SECFAULT handler will be called.
8. If one of rules 4, 5 or 6 apply but none of rules 1, 2, 3 and sec=true then MEMFAULT_S handler will be called.
9. If one of rules 4, 5 or 6 apply but none of rules 1, 2, 3 and sec=false then MEMFAULT_NS handler will be called.

10. Modification of sec to value true is only allowed for SG, FNC_RETURN, EXC_RETURN or HANDLER_S.
11. Modification of sec to value false is only allowed for BNS and EXC_RETURN.
12. Modification of nPriv_S to value false is only allowed when handlermode = true and sec = true.
13. Modification of nPriv_S to value true is only allowed when sec = true.
14. Modification of nPriv_NS to value false is only allowed when handlermode = true
    or (sec = true and nPriv_S = false).
15. Modification of handlermode to value true is only allowed for HANDLER_S or HANDLER_NS.
16. Modification of handlermode to value false is only allowed for EXC_RETURN.
17. Modification of nPriv_NS to value true is only allowed when privileged = true

Roles for management:

The parameter x designates any address.

- secure AF management: privileged = true and sec = true
- non-secure AF management: privileged = true

6.1.8.2 SFRs

Table 24 FDP_ACC.2/AF

FDP_ACC.2/AF	**Complete access control**
Hierarchical to	FDP_ACC.1
Dependencies	FDP_ACF.1
FDP_ACC.2.1/AF	The TSF shall enforce the Application Firewall access control policy[1] on subjects, objects and operations defined in 6.1.8.1[2] and all operations among subjects and objects covered by the SFP.
FDP_ACC.2.2/AF	The TSF shall ensure that all operations between any subject controlled by the TSF and any object controlled by the TSF are covered by an access control SFP.

[1] [assignment: access control SFP]

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

Table 25 FDP_ACF.1/AF

FDP_ACF.1/AF	**Security attribute based access control**
Hierarchical to	No other components.
Dependencies	FDP_ACC.1 FMT_MSA.3
FDP_ACF.1.1/AF	The TSF shall enforce the Application Firewall access control policy[1] to objects based on the following: The subjects, objects, operations and associated security attributes defined in 6.1.8.1[2].
FDP_ACF.1.2/AF	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: Rules defined in 6.1.8.1[3].
FDP_ACF.1.3/AF	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: None[4].
FDP_ACF.1.4/AF	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: None[5].

Table 26 FMT_MSA.3/AF

FMT_MSA.3/AF	**Static attribute initialisation**
Hierarchical to	No other components.
Dependencies	FMT_MSA.1 FMT_SMR.1
FMT_MSA.3.1/AF	The TSF shall enforce the Application Firewall access control policy[6] to provide restrictive[7] default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2/AF	The TSF shall allow the none[8] to specify alternative initial values to override the default values when an object or information is created.

Note: Restrictive means that the security attributes for all addresses are sec(x) = S

Table 27 FMT_MSA.1/AF/S

FMT_MSA.1/AF/S	**Management of security attributes**
Hierarchical to	No other components.
Dependencies	[FDP_ACC.1 or FDP_IFC.1] FMT_SMR.1

[1] [assignment: access control SFP]

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

[3] [assignment: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]

[4] [assignment: rules, based on security attributes, that explicitly authorize access of subjects to objects].

[5] [assignment: rules, based on security attributes, that explicitly deny access of subjects to objects].

[6] [assignment: access control SFP, information flow control SFP]

[7] [selection, choose one of: restrictive, permissive, [assignment: other property]]

[8] [assignment: the authorized identified roles]

FMT_MSA.1/AF/S	Management of security attributes
	FMT_SMF.1
FMT_MSA.1.1/AF/S	The TSF shall enforce the Application Firewall access control policy[1] to restrict the ability to modify[2] the security attributes PO(x), acc(x), sec(x), XN(x)[3] to secure AF management in case sec(x) = S[4].

Table 28 FMT_MSA.1/AF/NS

FMT_MSA.1/AF/NS	Management of security attributes
Hierarchical to	No other components.
Dependencies	[FDP_ACC.1 or FDP_IFC.1] FMT_SMR.1 FMT_SMF.1
FMT_MSA.1.1/AF/NS	The TSF shall enforce the Application Firewall access control policy[5] to restrict the ability to modify[6] the security attributes PO(x), acc(x), XN(x)[7] to secure or non-secure AF management in case sec(x) = NS[8].

Table 29 FMT_SMF.1/AF

FMT_SMF.1/AF	Specification of management functions
Hierarchical to	No other components.
Dependencies	No dependencies.
FMT_SMF.1.1/AF	The TSF shall be capable of performing the following management functions: Modification of the security attributes PO(x), acc(x), sec(x), XN(x)[9].

Table 30 FMT_SMR.1/AF

FMT_SMR.1/AF	Security Roles
Hierarchical to	No other components.
Dependencies	FIA_UID.1
FMT_SMR.1.1/AF	The TSF shall maintain the roles secure AF management and non-secure AF management[10].
FMT_SMR.1.2/AF	The TSF shall be able to associate users with roles.

[1] [assignment: access control SFP(s), information flow control SFP(s)]
[2] [selection: change_default, query, modify, delete, [assignment: other operations]]
[3] [assignment: list of security attributes]
[4] [assignment: the authorized identified roles]
[5] [assignment: access control SFP(s), information flow control SFP(s)]
[6] [selection: change_default, query, modify, delete, [assignment: other operations]]
[7] [assignment: list of security attributes]
[8] [assignment: the authorized identified roles]
[9] [assignment: list of management functions to be provided by the TSF]
[10] [assignment: the authorised identified roles]

6.1.9 Authentication of the Security IC

The TOE shall implement the Package “Authentication of the Security IC” from [PP0084], ch. 7.2.

Table 31 FIA_API.1

FIA_API.1	**Authentication Proof of Identity**
Hierarchical to	No other components.
Dependencies	No dependencies.
FIA_API.1.1	The TSF shall provide a authentication mechanism according to [ISO9798_2] section 7.3.3, Mechanism MUT.CR-Three-pass authentication[1] to prove the identity of the TOE to an external entity.

Note: FIA_API is only available, if the Flash Loader is active.

6.1.10 Flash loader

The TOE provides a Flash Loader to download user data into the NVM, either during production of the TOE or at customer site. This TOE shall support both Loader packages from [PP0084] section 7.3.

- Package 1: Loader dedicated for usage in secured environment only.
- Package 2: Loader dedicated for usage by authorized users only.

The SFRs FDP_UCT.1 and FDP_UIT.1 are specified in [PP0084].

Table 32 FMT_LIM.1/Loader

FMT_LIM.1/Loader	**Limited Capabilities - Loader**
Hierarchical to	No other components.
Dependencies	FMT_LIM.2
FMT_LIM.1.1/Loader	The TSF shall be designed and implemented in a manner that limits its capabilities so that in conjunction with “Limited availability (FMT_LIM.2)” the following policy is enforced: Deploying Loader functionality after permanent deactivation[2] does not allow stored user data to be disclosed or manipulated by unauthorized user.

Table 33 FMT_LIM.2/Loader

FMT_LIM.2/Loader	**Limited availability - Loader**
Hierarchical to	No other components.
Dependencies	FMT_LIM.1
FMT_LIM.2.1/Loader	The TSF shall be designed in a manner that limits its availability so that in conjunction with “Limited capabilities (FMT_LIM.1)” the following policy is enforced: The TSF prevents deploying the Loader functionality after permanent deactivation[3].

[1] [assignment: authentication mechanism]
[2] [assignment: action]
[3] [assignment: action]

Note: The User Guidance for this TOE requires the Flash Loader to be permanently deactivated prior delivery to the end user (Phase 7).

Table 34 FTP_ITC.1

FTP_ITC.1	Inter-TSF trusted channel
Hierarchical to	No other components.
Dependencies	No dependencies.
FTP_ITC.1.1	The TSF shall provide a communication channel between itself and Administrator User or Download Operator User and Image Provider[1] that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.
FTP_ITC.1.2	The TSF shall permit another trusted IT product to initiate communication via the trusted channel.
FTP_ITC.1.3	The TSF shall initiate communication via the trusted channel for deploying Loader for downloading User Data and modification of authentication keys[2].

Note: The download operation is authenticated by the Administrator User or the Download Operator User but the download image may be encrypted and authenticated by a different role. This role is called the "Image Provider". Thus, the download operation provides in effect a trusted channel between the Image Provider and the Flash Loader.

Table 35 FDP_ACC.1/Loader

FDP_ACC.1/Loader	Subset access control - Loader
Hierarchical to	No other components.
Dependencies	FDP_ACF.1
FDP_ACC.1.1/Loader	The TSF shall enforce the Loader SFP on (1) the subjects Administrator User, Download Operator User and Image Provider[3], (2) the objects user data in NVM[4], (3) the operation deployment of Loader.

Table 36 FDP_ACF.1/Loader

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

[1] [assignment: users authorized for using the Loader]
[2] [assignment: rules]
[3] [assignment: authorized roles for using Loader]
[4] [assignment: memory areas]

FDP_ACF.1/Loader	Security attribute based access control - Loader
	(1) the subjects Administrator User, Download Operator User and Image Provider[1] with security attributes None[2]. (2) the objects user data in NVM[3] with security attributes None[4].
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: The authenticated Administrator User or authenticated Download Operator User can replace the user data by new user data when the new user data is authorized by the Image Provider[5].
FDP_ACF.1.3/Loader	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: None[6].
FDP_ACF.1.4/Loader	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: None[7].

Note: The Image provider authenticates with the flash loader implicitly by providing a correctly signed and encrypted download image. An Image provider authentication must always be preceded by an Administrator User or Download Operator User authentication.

6.1.10.1 SFRs added in this ST

The following SFRs have been added to the SFRs from Flash Loader package 2 of [PP0084] in order to describe the management of the various Flash Loader authentication keys.

Table 37 FMT_MTD.1/Loader

FMT_MTD.1/Loader	Management of TSF data
Hierarchical to	No other components.
Dependencies	FMT_SMR.1 FMT_SMF.1
FMT_MTD.1.1/Loader	The TSF shall restrict the ability to modify, delete[8] the Authentication keys for Administrator User, Download Operator User and Image Provider[9] to Administrator User, Download Operator User[10].

Note: The Administrator User can manage the keys for Administration User, Download Operator User and Image Provider.
The Download Operator User can delete the key for Image Provider and Download Operator and modify keys for the Download Operator User only.
The image provider cannot modify any keys or perform authentication with the Flash Loader. It can only build encrypted and authenticated loadable images.

[1] [assignment: authorized roles for using Loader]
[2] [assignment: SFP relevant security attributes, or named groups of SFP relevant security attributes]
[3] [assignment: memory areas]
[4] [assignment: SFP-relevant security attributes, or named groups of SFP-relevant security attributes]
[5] [assignment: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]
[6] [assignment: rules, based on security attributes, that explicitly authorise access of subjects to objects].
[7] [assignment: rules, based on security attributes, that explicitly deny access of subjects to objects]
[8] [selection: change_default, query,modify, delete, clear, [assignment: other operations]]
[9] [assignment: list of TSF data]
[10] [assignment: the authorised identified roles]

Table 38 FMT_SMR.1/Loader

FMT_SMR.1/Loader	Security roles
Hierarchical to	No other components.
Dependencies	FIA_UID.1
FMT_SMR.1.1/Loader	The TSF shall maintain the roles Administrator User, Download Operator User, Image Provider[1].
FMT_SMR.1.2/Loader	The TSF shall be able to associate users with roles.

Note: Image provider is the role who maintains the key which is used to encrypt and integrity protect the download image.

Table 39 FMT_SMF.1/Loader

FMT_SMF.1/Loader	Specification of Management Functions
Hierarchical to	No other components.
Dependencies	No dependencies.
FMT_SMF.1.1/Loader	The TSF shall be capable of performing the following management functions: Change Key, Invalidate Key[2].

Note: "Change Key" of this SFR means the "modify" operations from SFR FMT_MTD.1/Loader, "Invalidate Key" of this SFR means the "delete" operation from SFR FMT_MTD.1/Loader.

Table 40 FIA_UID.2/Loader

FIA_UID.2/Loader	User identification before any action
Hierarchical to	FIA_UID.1
Dependencies	No dependencies.
FIA_UID.2.1/Loader	The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

6.1.11 Crypto Suite

This chapter defines the SFRs related to the optional Crypto Suites. Please note that the SFRs from this chapter are only applicable if the optional Crypto Suites is delivered.

6.1.11.1 AES

This section describes AES ciphers provided by the Crypto Suite.

[1] [assignment: the authorised identified roles]
[2] [assignment: list of management functions to be provided by the TSF]

Table 41 FCS_COP.1/CS/AES/<iter>

FCS_COP.1/CS/AES/<iter>	Cryptographic operation
Dependencies	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4
FCS_COP.1.1/CS/AES/<iter>	The TSF shall perform [assignment: list of cryptographic operations] in accordance with a specified cryptographic algorithm [assignment: cryptographic algorithms] and cryptographic key sizes [assignment: cryptographic key sizes] that meet the following: [assignment: list of standards].

The operations performed in this SFR are defined in the following table. Please note that **<iter>** is a placeholder for different SFR iterations defined in the first column.

Table 42 Cryptographic table for FCS_COP.1/CS/AES/<iter>

<iter>	[assignment: list of cryptographic operations]	[assignment: cryptographic algorithms]	[assignment: cryptographic key sizes]	[assignment: list of standards]
ENC	encryption and decryption	AES ECB, CBC mode	128,192,256 bits	[FIPS 197] [SP 800-38A]
MAC	MAC generation	AES CMAC mode	128,192,256 bits	[FIPS 197] [SP 800-38B]

6.1.11.2 TDES

This section describes DES ciphers provided by the Crypto Suite.

Table 43 FCS_COP.1/CS/TDES/<iter>

FCS_COP.1/CS/TDES/<iter>	Cryptographic operation
Dependencies	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4
FCS_COP.1.1/CS/TDES/<iter>	The TSF shall perform [assignment: list of cryptographic operations] in accordance with a specified cryptographic algorithm [assignment: cryptographic algorithms] and cryptographic key sizes [assignment: cryptographic key sizes] that meet the following: [assignment: list of standards].

The operations performed in this SFR are defined in the following table. Please note that **<iter>** is a placeholder for different SFR iterations defined in the first column.

Table 44 Cryptographic table for FCS_COP.1/CS/TDES/<iter>

<iter>	[assignment: list of cryptographic operations]	[assignment: cryptographic algorithms]	[assignment: cryptographic key sizes]	[assignment: list of standards]
ENC	encryption and decryption	TDES ECB, CBC mode	112, 168 bits	[SP 800-67] [SP 800-38A]
MAC	MAC calculation	TDES Retail MAC	112 bits	[SP 800-67] [ISO9797_1]

Note: The Crypto Suite also provides hardened single DES (56 bits) algorithms. This algorithm is also hardened and evaluated but is not part of the formal certification scope.

6.1.11.3 FFC

This section describes Finite Field algorithms provided by the Crypto Suite.

Table 45 FCS_COP.1/CS/FFC/<iter>

FCS_COP.1/CS/FFC/<iter>	Cryptographic operation
Hierarchical to	No other components.
Dependencies	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4
FCS_COP.1.1/CS/FFC/<iter>	The TSF shall perform [assignment: list of cryptographic operations] in accordance with a specified cryptographic algorithm [assignment: cryptographic algorithms] and cryptographic key sizes [assignment: cryptographic key sizes] that meet the following: [assignment: list of standards].

The operations performed in this SFR are defined in the following table. Please note that **<iter>** is a placeholder for different SFR iterations defined in the first column.

Table 46 Cryptographic table for FCS_COP.1/CS/FFC/<iter>

<iter>	[assignment: list of cryptographic operations]	[assignment: cryptographic algorithms]	[assignment: cryptographic key sizes]	[assignment: list of standards]
DH	key agreement	Finite field Diffie-Hellman	1024-2048 bits	[PKCS#3], ch 7.2

6.1.11.4 RSA

This section describes RSA related algorithms provided by the Crypto Suite.

Table 47 FCS_COP.1/CS/RSA/<iter>

FCS_COP.1/CS/RSA/<iter>	Cryptographic operation
Hierarchical to	No other components.

FCS_COP.1/CS/RSA/<iter>	Cryptographic operation
Dependencies	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4
FCS_COP.1.1/CS/RSA/<iter>	The TSF shall perform [assignment: list of cryptographic operations] in accordance with a specified cryptographic algorithm [assignment: cryptographic algorithms] and cryptographic key sizes [assignment: cryptographic key sizes] that meet the following: [assignment: list of standards].

The operations performed in this SFR are defined in the following table. Please note that **<iter>** is a placeholder for different SFR iterations defined in the first column.

Table 48 Cryptographic table for FCS_COP.1/CS/RSA/<iter>

<iter>	[assignment: list of cryptographic operations]	[assignment: cryptographic algorithms]	[assignment: cryptographic key sizes]	[assignment: list of standards]
ENC	encryption	RSAEP	1024-4224 bits	[PKCS#1], ch. 5.1.1
DEC	decryption	RSADP	1024-2112 bits	[PKCS#1], ch. 5.1.2, 2a
DEC_CRT	decryption	RSADP(CRT)	1024-4224 bits	[PKCS#1], ch. 5.1.2, 2b
SIG	signature generation	RSASP1	1024-2112 bits	[PKCS#1], ch. 5.2.1, 2a
SIG_CRT	signature generation	RSASP1(CRT)	1024-4224 bits	[PKCS#1], ch. 5.2.1, 2b
VER	signature verification	RSAVP1	1024-4224 bits	[PKCS#1], ch. 5.2.2

6.1.11.5 ECC

This section describes ECC related algorithms provided by the Crypto Suite.

Table 49 FCS_COP.1/CS/ECC/<iter>

FCS_COP.1/CS/ECC/<iter>	Cryptographic operation
Hierarchical to	No other components.
Dependencies	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4
FCS_COP.1.1/CS/ECC/<iter>	The TSF shall perform [assignment: list of cryptographic operations] in accordance with a specified cryptographic algorithm [assignment: cryptographic algorithms] and cryptographic key sizes [assignment: cryptographic key sizes] that meet the following: [assignment: list of standards].

The operations performed in this SFR are defined in the following table. Please note that **<iter>** is a placeholder for different SFR iterations defined in the first column.

Table 50 Cryptographic table for FCS_COP.1/CS/ECC/<iter>

<iter>	**[assignment: list of cryptographic operations]**	**[assignment: cryptographic algorithms]**	**[assignment: cryptographic key sizes]**	**[assignment: list of standards]**
ECDSA_SIG	EC signature generation	ECDSA	160-521 bits	[FIPS 186-4], ch. 6.4 [FIPS 186-5], ch. 6.4.1
ECDSA_VER	EC signature verification	ECDSA	160-521 bits	[FIPS 186-4], ch. 6.4 [FIPS 186-5], ch. 6.4.2
ECDH	key agreement	ECDH	160-521 bits	[SP 800-56A], ch. 5.7.1.2

Note: [FIPS 186-5] requires key size ≥ 224 bits.

Note: The ECDH operation returns the y-coordinate in addition to the x-coordinate.

Note: The ECDSA_SIG and ECDSA_VER operations will not hash the message input.

The following table lists the certified elliptic curves.

Table 51 Certified elliptic curves

Curve	**Reference**
NIST curves over prime fields	[FIPS 186-5]
Brainpool curves	[RFC 5639]

Note: The Crypto Suite supports any elliptic curve in short Weierstrass form up to 521 bits.

Table 52 FCS_CKM.1/CS/ECC

FCS_CKM.1/CS/ECC	**Cryptographic operation**
Hierarchical to	No other components.
Dependencies	FCS_CKM.2 or FCS_COP.1] FCS_CKM.4
FCS_CKM.1.1/CS/ECC	The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm EC key generation[1] and specified cryptographic key sizes of 160-521 bits[2] that meet the following: [FIPS 186-4], B.4.1 or [FIPS 186-5], A.2.1[3].

[1] [assignment: cryptographic key generation algorithm]
[2] [assignment: cryptographic key sizes]
[3] [assignment: list of standards]

Note: [FIPS 186-5] requires key size ≥ 224 bits.

6.1.11.6 Hash

This section describes hash related algorithms provided by the Crypto Suite.

Table 53 FCS_COP.1/CS/Hash/<iter>

FCS_COP.1/CS/Hash/<iter>	Cryptographic operation
Hierarchical to	No other components.
Dependencies	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4
FCS_COP.1.1/CS/Hash/<iter>	The TSF shall perform [assignment: list of cryptographic operations] in accordance with a specified cryptographic algorithm [assignment: cryptographic algorithms] and cryptographic key sizes [assignment: cryptographic key sizes] that meet the following: [assignment: list of standards].

The operations performed in this SFR are defined in the following table. Please note that **<iter>** is a placeholder for different SFR iterations defined in the first column.

Table 54 Cryptographic table for FCS_COP.1/CS/Hash/<iter>

<iter>	[assignment: list of cryptographic operations]	[assignment: cryptographic algorithms]	[assignment: cryptographic key sizes]	[assignment: list of standards]
SHA1	Hash digest	SHA-1	none	[FIPS 180-4]
SHA2	Hash digest	SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256	none	[FIPS 180-4]

6.1.11.7 Random

This section describes random number generation provided by the Crypto Suite.

Table 55 FCS_RNG.1/CS/PTG2

FCS_RNG.1/CS/PTG2	Random Number Generation
Hierarchical to	No other components.
Dependencies	No dependencies.
FCS_RNG.1.1/CS/PTG2	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.

FCS_RNG.1/CS/PTG2	Random Number Generation
	(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[1]. (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[2]. 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/CS/PTG2	The TSF shall provide 32-bit numbers[3] that meet (PTG.2.6) Test procedure A (None)[4] 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.

Note: The PTG.2 API provided by the Crypto Suite is only a wrapper to the hardware-provided PTG.2 defined in section 6.1.1.

Table 56 FCS_RNG.1/CS/PTG3

FCS_RNG.1/CS/PTG3	Cryptographic operation
Hierarchical to	No other components.
Dependencies	No dependencies.
FCS_RNG.1.1/CS/PTG3	The TSF shall provide a hybrid physical random number generator that implements: (PTG.3.1) 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.3.2) If a total failure of the entropy source occurs while the RNG is being operated, the RNG prevents the output of any internal random

[1] [selection: prevents the output of any internal random number that depends on some raw random numbers that have been generated after the total failure of the entropy source, generates the internal random numbers with a post-processing algorithm of class DRG.2 as long as its internal state entropy guarantees the claimed output entropy]

[2] [selection: externally, at regular intervals, continuously, applied upon specified internal events]

[3] [selection: bits, octets of bits, numbers [assignment: format of the numbers]]

[4] [assignment: additional standard test suites]

FCS_RNG.1/CS/PTG3	Cryptographic operation
	number that depends on some raw random numbers that have been generated after the total failure of the entropy source[1]. (PTG.3.3) 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 and the seeding of the DRG.3 post-processing algorithm have been finished successfully or when a defect has been detected. (PTG.3.4) The online test procedure shall be effective to detect non-tolerable weaknesses of the random numbers soon. (PTG.3.5) The online test procedure checks the raw random number sequence. It is triggered continuously[2]. 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. (PTG.3.6) The algorithmic post-processing algorithm belongs to Class DRG.3 with cryptographic state transition function and cryptographic output function, and the output data rate of the post-processing algorithm shall not exceed its input data rate.
FCS_RNG.1.2/CS/PTG3	The TSF shall provide numbers in 32-bit packets[3] that meet: (PTG.3.7) Statistical test suites cannot practically distinguish the internal random numbers from output sequences of an ideal RNG. The internal random numbers must pass test procedure A (none)[4]. (PTG.3.8) The internal random numbers shall use PTRNG of class PTG.2 as random source for the post processing[5].

6.2 Security assurance requirements

In the following Table 57, the security assurance requirements and compliance rationale for augmented refinements are given.

Table 57 SAR list and refinements

SAR	Refinement
ADV_ARC.1	Refined in [PP0084]
ADV_FSP.5	The refinement of ADV_FSP.4 from [PP0084] can also be applied to the assurance level EAL 6 comprising ADV_FSP.5. The assurance component ADV_FSP.4 is extended to ADV_FSP.5 with aspects regarding the level of description. ADV_FSP.5 requires a semi-formal description in addition. The refinement is still valid.

[1] [selection: prevents the output of any internal random number that depends on some raw random numbers that have been generated after the total failure of the entropy source, generates the internal random numbers with a post-processing algorithm of class DRG.3 as long as its internal state entropy guarantees the claimed output entropy]

[2] [selection: externally, at regular intervals, continuously, upon specified internal events]

[3] [assignment: format of the numbers]]

[4] [assignment: additional test suites]

[5] [selection: use PTRNG of class PTG.2 as random source for the post-processing, have [assignment: work factor], require [assignment: guess work]]

SAR	Refinement
ADV_IMP.2	The refinement of ADV_IMP.1 in [PP0084] requires the evaluator to check for completeness. In case of ADV_IMP.2 the entire implementation representation has to be provided anyhow. A check for completeness is also applicable in case the entire implementation representation is provided.
ADV_INT.3	No refinement
ADV_TDS.5	No refinement
ADV_SPM.1	No refinement
AGD_OPE.1	Refined in [PP0084]
AGD_PRE.1	Refined in [PP0084]
ALC_CMC.5	The refinement of ALC_CMC.4 from [PP0084] details how configuration management has to be also applied to production. This is also applicable for ALC_CMC.5. ALC_CMC.5 is not specifically focused on production.
ALC_CMS.5	The refinement of ALC_CMS.4 from [PP0084] can also be applied to the assurance level EAL 6 comprising ALC_CMS.5. The assurance package ALC_CMS.4 is extended to ALC_CMS.5 with aspects regarding the configuration control system for the TOE. The refinement is still valid.
ALC_DEL.1	Refined in [PP0084]
ALC_DVS.2	Refined in [PP0084]
ALC_FLR.1	No refinement
ALC_LCD.1	No refinement
ALC_TAT.3	No refinement
ASE_CCL.1	No refinement
ASE_ECD.1	No refinement
ASE_INT.1	No refinement
ASE_OBJ.2	No refinement
ASE_REQ.2	No refinement
ASE_SPD.1	No refinement
ASE_TSS.1	No refinement
ATE_COV.3	The refinement of ATE_COV.2 in [PP0084] clarifies how to deal with testing of security mechanisms for physical protection. It further requests the TOE to be tested under different operating conditions. These refinements are also applicable for ATE_COV.3, which requires complete TSFI coverage.
ATE_DPT.3	No refinement
ATE_FUN.2	No refinement
ATE_IND.2	No refinement
AVA_VAN.5	Refined in [PP0084]

6.2.1 Security Policy Model (SPM) details

[CC3] requires in ADV_SPM.1.1D to define the not modelled SFRs.

The rational for the excluded SFRs are as follows:

- SFRs for cryptographic services are not modelled by convention.

- SFRs for physical functions cannot be logically modelled.
- SFRs for internal functions have no visible logical interface.

The developer shall provide a formal security policy model for the SFRs of this ST with the exception of the SFRs from the following table[1].

Table 58 SFRs excluded from SPM

SFR	Reason for exclusion
FCS_RNG.1/*	cryptographic services
FCS_COP.1/*	cryptographic services
FCS_CKM.1/*	cryptographic services
FCS_CKM.4/*	cryptographic services
FPT_TST.1	physical function
FRU_FLT.2	physical function
FPT_FLS.1	physical function
FPT_PHP.3	physical function
FDP_SDC.1	physical function
FDP_ITT.1	internal function
FPT_ITT.1	Internal function
FDP_IFC.1	Internal function

Note: A star "" means all iterations of that SFR*

6.3 Security requirements rationale

6.3.1 Rationale for the Security Functional Requirements

The security requirements rationale identifies the modifications and additions made to the rationale presented in [PP0084].

6.3.1.1 Additional SFRs related to O.Firewall

Table 59 Rationale for SFRs related to O.Firewall

SFR	Rationale
FDP_ACC.2/AF	The SFR with the respective SFP require the implementation of an area-based memory access control.
FDP_ACF.1/AF	The SFR allows the TSF to enforce access to objects within the respective SFP based on security attributes and defines these attributes and defines the rules based on these attributes that enable explicit decisions.
FMT_MSA.3/AF	The SFR requires that the TOE provides default values for the security attributes used in the SFP. Because the TOE is a hardware platform, these default values are generated by the reset procedure.

[1] [assignment: list of policies that are formally modelled]

SFR	Rationale
FMT_MSA.1/AF/S	The SFR requires that authorized users can manage TSF attributes. It ensures that the access control attributes associated to secure addresses can be managed only by code running in secure and privilege mode.
FMT_MSA.1/AF/NS	The SFR requires that authorized users can manage TSF attributes. It ensures that the access control attributes associated to non-secure addresses can be managed by code running in secure or non-secure privilege mode.
FMT_SMF.1/AF	The SFR is used for the specification of the management functions to be provided by the TOE. Being a hardware platform, the TOE allows the management of the security attributes by making the hardware registers accessible to software to enable modification.
FMT_SMR.1/AF	This SFR defines the roles used for management of the security attributes. The roles are defined by the security attribute of the fetch address of the CPU instruction.

6.3.1.2 Additional SFRs related to O.Ctrl_Auth_Loader

Table 60 Rationale for additional SFRs related to O.Ctrl_Auth_Loader

SFR	Rationale
FMT_MTD.1/Loader	This SFR requires that the TOE provides management functions for modification and deletion of authentication keys.
FMT_SMR.1/Loader	This SFR requires that the roles to management keys are defined
FMT_SMF.1/Loader	This SFR requires that the key management functions are defined
FIA_UID.2/Loader	This SFR requires that management functions can only be executed by authorized roles.

6.3.1.3 Additional SFRs related to O.Phys-Manipulation

The FPT_TST.1 component allows that particular parts of the security mechanisms and functions provided by the TOE can be tested after TOE delivery. This feature is important to detect direct physical manipulations by a FIB device in order to disable the alarm system of the chip.

6.3.1.4 Additional SFRs related to O.AES

The optional Crypto Suites provides the AES chaining modes ECB and CBC. The associated SFR is FCS_COP.1/CS/AES/ENC from chapter 6.1.11.1.

6.3.1.5 Additional SFRs related to O.AES-CMAC

The optional Crypto Suites provides the AES CMAC. The associated SFR is FCS_COP.1/CS/AES/MAC from chapter 6.1.11.1.

6.3.1.6 Additional SFRs related to O.TDES

The optional Crypto Suites provides the TDES chaining modes ECB and CBC. The associated SFR is FCS_COP.1/CS/TDES/ENC from chapter 6.1.11.2.

6.3.1.7 Additional SFRs related to O.TDES-RMAC

The optional Crypto Suites provides the TDES Retail Mac. The associated SFR is FCS_COP.1/CS/TDES/MAC from chapter 6.1.11.2.

6.3.1.8 Additional SFRs related to O.FFC

The optional Crypto Suites provides Finite Field cryptographic services. The associated SFRs are defined in chapter 6.1.11.3.

6.3.1.9 Additional SFRs related to O.RSA

The optional Crypto Suites provides RSA functions. The associated SFRs are defined in chapter 6.1.11.4.

6.3.1.10 Additional SFRs related to O.ECC

The optional Crypto Suites provides ECC functions. The associated SFRs are defined in chapter 6.1.11.5.

6.3.1.11 Additional SFRs related to O.Hash

The optional Crypto Suites provides Hash functions. The associated SFRs are defined in chapter 6.1.11.6.

6.3.1.12 Additional SFRs related to O.RND

The optional Crypto Suites provides Random generating functions. The associated SFRs are defined in chapter 6.1.11.7.

6.3.2 Dependencies of Security Functional Requirements

The dependencies of the SFRs which are defined in [PP0084] are resolved in [PP0084], ch. 6.3.2.
The following table lists the dependencies of the additional SFRs which are defined in this ST.

Table 61 Dependencies of SFRs

SFR	Dependencies	Rationale
FDP_ACC.2/AF	FDP_ACF.1	Fulfilled by FDP_ACF.1/AF
FDP_ACF.1/AF	FDP_ACC.1	Fulfilled by FDP_ACC.2/AF, which is hierarchically higher
FMT_MSA.3/AF	FMT_MSA.1	Fulfilled by FMT_MSA.1/AF/S and FMT_MSA.1/AF/NS
	FMT_SMR.1	Fulfilled by FMT_SMF.1/AF
FMT_MSA.1/AF/S FMT_MSA.1/AF/NS	FDP_ACC.1 or FDP_IFC.1	Fulfilled by FDP_ACC.2/AF, which is hierarchically higher
	FMT_SMR.1	Fulfilled by FMT_SMF.1/AF
	FMT_SMF.1	Fulfilled by FMT_SMF.1/AF
	FMT_SMR.1	Fulfilled by FMT_SMF.1/AF
	FMT_SMF.1	Fulfilled by FMT_SMF.1/AF
FMT_SMR.1/AF	FIA_UID.1	The dependency is satisfied, because the role is identified by the execution context of the processor.
FMT_SMF.1/AF	None	No dependency
FDP_ACC.1/Loader	FDP_ACF.1	Fulfilled by FDP_ACF.1/Loader
FDP_ACF.1/Loader	FMT_MSA.3	Not applicable, because there are no security attributes defined

SFR	Dependencies	Rationale
FMT_MTD.1/Loader	FMT_SMR.1	Fulfilled by FMT_SMR.1/Loader
	FMT_SMF.1	Fulfilled by FMT_SMF.1/Loader
FMT_SMR.1/Loader	FIA_UID.1	Fulfilled by FIA_UID.2/Loader
FMT_SMF.1/Loader	None	No dependency
FIA_UID.2/Loader	None	No dependency
FPT_TST.1	None	No dependency
FCS_CKM.4/AES	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	The TOE does not provide services to generate or import symmetric keys. This will be done by the embedded software for the composite TOE.
FCS_COP.1/CS/AES/<iter>	FCS_CKM.4	Fulfilled by FCS_CKM.4/AES
	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	The TOE does not provide services to generate or import symmetric keys. This will be done by the embedded software for the composite TOE.
FCS_COP.1/CS/TDES/<iter>	FCS_CKM.4	Fulfilled by FCS_CKM.4/TDES
	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	The TOE does not provide services to generate or import symmetric keys. This will be done by the embedded software for the composite TOE.
FCS_CKM.4/TDES	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	The TOE does not provide services to generate or import symmetric keys. This will be done by the embedded software for the composite TOE.
FCS_COP.1/CS/FFC/<iter>	FCS_CKM.4	The TOE does not provide services to destroy FFC keys. This will be done by the embedded software for the composite TOE.
	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	The TOE does not provide services to generate or import FFC keys. This will be done by the embedded software for the composite TOE.
FCS_COP.1/CS/RSA/<iter>	FCS_CKM.4	The TOE does not provide services to destroy RSA keys. This will be done by the embedded software for the composite TOE.
	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	The TOE does not provide services to generate or import RSA keys. This will be done by the embedded software for the composite TOE.
FCS_COP.1/CS/ECC/<iter>	FCS_CKM.4	The TOE does not provide services to destroy ECC keys. This will be done by the embedded software for the composite TOE.
	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	Fulfilled by FCS_CKM.1/CS/ECC/<iter>
FCS_CKM.1/CS/ECC/<iter>	FCS_CKM.2 or FCS_COP.1]	Fulfilled by FCS_COP.1/CS/ECC/<iter>
	FCS_CKM.4	The TOE does not provide services to destroy ECC keys. This will be done by the embedded software for the composite TOE.
FCS_COP.1/CS/Hash/<iter>	FCS_CKM.4	n/A, as a hash function does not use keys.

SFR	Dependencies	Rationale
	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	n/A, as a hash function does not use keys.
FCS_RNG.1/CS/PTG2	None	No dependency
FCS_RNG.1/CS/PTG3	None	No dependency

6.3.3 Rationale of the Assurance Requirements

The TOE is a typical security IC as defined in [PP0084]. The rationale for EAL level and augmentation is as follows.

An assurance level EAL6 with the augmentations ALC_FLR.1 is required for this type of TOE since it is intended to defend against highly sophisticated attacks without a protective environment. This evaluation assurance package was selected to permit a developer to gain maximum assurance from positive security engineering based on good commercial practices. To provide a meaningful level of assurance that the TOE provides an adequate level of defence against such attacks, the evaluators should have access to all information regarding the TOE including the TSF internals, the low-level design and source code including the testing of the modular design. Additionally, the mandatory technical document [JIL] shall be taken as a basis for the vulnerability analysis of the TOE.

7 TOE Summary Specification (ASE_TSS)

The product overview is given in section 1.3.1. The Security Features are described below and the relation to the security functional requirements is shown. The TOE is equipped with the following security features to meet the security functional requirements:

Table 62 TOE Security Features

Security Feature	Description
SF_DPM	Device Phase Management
SF_PS	Protection against Snooping
SF_PMA	Protection against Modification Attacks
SF_PLA	Protection against Logical Attacks
SF_HC	Hardware provided Cryptography
SF_CS	Crypto Suite Services

7.1 SF_DPM: Device Phase Management

The life cycle of the TOE is split up into several phases (see [PP0084], ch. 1.2.3). Chip development and production (phase 2, 3, 4) and final use (phases 4-7) is a rough split-up from the TOE point of view. These phases are implemented in the TOE as test mode (phase 3) and user mode (phases 4-7). In addition, a chip identification mode exists which is active in all phases. The chip identification data (O.Identification) is stored in a non-modifiable configuration page area of the non-volatile memory. Further TOE configuration data is stored in the same area. In addition, user initialization data can be stored in the NVM during the production phase as well. During this first data programming, the TOE is still in the secured environment and in test mode.

The covered security functional requirement is FAU_SAS.1 "Audit storage".

During start-up of the TOE the decision for one of the various operation modes is taken dependent on phase identifiers. The decision of accessing a certain mode is defined as phase entry protection. The phases follow also a defined and protected sequence. The sequence of the phases is protected by means of authentication.

The covered security functional requirements are FMT_LIM.1 and FMT_LIM.2.

During the production phase (phase 3 and 4) or after the delivery to the customer (phase 5 or phase 6), the TOE provides the possibility to download a user specific encryption key and user code and data into the empty (erased) NVM area as specified by the associated control information of the Flash Loader software. Alternatively in case the user has ordered TOE derivatives without Flash Loader, software download by the user (phase 5 or phase 6) is disabled and all user data of the embedded software is stored on the TOE at Infineon premises. In case the user has ordered the TOE derivatives with Flash Loader enabled, the Flash Loader may either be received in a way, which requires an authentic Pinletter and authentication afterwards, or it may be received in a state, which immediately requires successful mutual authentication. The Pinletter process can exchange the default authentication key. Successful authentication is required before being able to use the download functionality of the Flash Loader. Once authenticated, the functionality to exchange the Flash Loader keys depending on the user's identity is enabled. One of the keys, which can be exchanged is the Image Provider key. This key is used to decrypt and verify the integrity protected and encrypted download image. The authenticated user may also invalidate authentication keys depending on the user's identity. After finishing the download operation, the Flash Loader must be permanently deactivated prior delivery to the end user, so that no further load operation with the Flash Loader is possible. The Flash Loader uses AES CCM mode [SP 800-38C] for encryption and integrity protection of payload and for authentication. For key usage diversification, the Flash Loader uses key derivation according to [SP 800-108].

The covered security functional requirements are FMT_LIM.1/Loader, FMT_LIM.2/Loader, FTP_ITC.1, FDP_UCT.1, FDP_UIT.1, FDP_ACC.1/Loader, FDP_ACF.1/Loader, FMT_MTD.1/Loader, FMT_SMR.1/Loader, FMT_SMF.1/Loader, FIA_UID.2/Loader and FIA_API.1.

Note that the SFRs FTP_ITC.1, FDP_UCT.1, FDP_UIT.1, FDP_ACC.1/Loader, FDP_ACF.1/Loader, FMT_MTD.1/Loader, FMT_SMR.1/Loader, FMT_SMF.1/Loader, FIA_UID.2/Loader and FIA_API.1 are only part of the TOE if the flash loader is active.

Each operation phase is protected by means of authentication and encryption.

The covered security functional requirements are FDP_ITT.1 and FPT_ITT.1.

7.2 SF_PS: Protection against Snooping

All contents of the memories RAM, ROM and NVM of the TOE are encrypted on chip to protect them against data analysis. The encryption of the memory content is done by the MCICE using a proprietary cryptographic algorithm. A complex key management and address scrambling provides protection against cryptographic analysis attacks. All security relevant transfers via the peripheral bus are dynamically masked and thus protected against readout and analysis. Leakage of data dependent code execution can be reduced by employing specific hardware features.

The covered security functional requirements are FDP_SDC.1, FDP_IFC.1, FPT_PHP.3, FPT_ITT.1, FPT_FLS.1 and FDP_ITT.1.

Most components of the design are synthesized to disguise allocation of elements to certain modules of the IC. Physical regularity of the logic functions is thereby removed. The covered security functional requirement is FPT_PHP.3.

A further protective design method used is security optimized wiring. Certain security-critical wires have been identified and protected by special routing measures against probing. Additionally specific signal lines, required to operate the device, are embedded into shield lines of the chip to prevent successful probing. The covered security functional requirements are FPT_PHP.3, FPT_ITT.1, FDP_ITT.1 and FDP_IFC.1.

A low system frequency sensor FSE is implemented to prevent the TOE from single stepping. The sensor is tested by the User Mode Security Life Control UMSLC. The UMSLC library provides some wrapper functionality around the UMSLC hardware part containing measures against fault attacks. The covered security functional requirements are FPT_PHP.3 and FPT_FLS.1.

7.3 SF_PMA: Protection against Modifying Attacks

The TOE has implemented a dual CPU running in lockstep mode and registers protected with 32-bit EDC. This mechanism reliably detects attacks on the code flow and data processed by the CPU. In the case of a detected attack, the TOE enters the secure state.

The TOE is equipped with a 28-bit EDC in RAM, a 28 bit EDC in NVM and a 32 bit EDC in ROM, which is realized in the MCICE peripheral. The EDC detects detect single- and multi-bit errors. In the case of an EDC error, the TOE enters the secure state.

The covered security functional requirements are FRU_FLT.2, FPT_PHP.3 and FDP_SDI.2.

A life test on internal security features is provided – it is called User Mode Security Life Control (UMSLC), which checks alarm lines for correct operation. This test can be triggered by user software during normal operation or via the UMSLC lib. If physical manipulation or a physical probing attack is detected, the TOE enters the secure state (as defined in chapter 6.1.4). To further decrease the risk of manipulation and tampering of the detection system a redundant alarm propagation and system deactivation is provided.

The covered security functional requirements are FPT_FLS.1, FPT_PHP.3 and FPT_TST.1

The Instruction Stream Signature Checking (ISS) calculates a hash over all executed instructions and automatically checks the correctness of this hash value. If the code execution follows an illegal path an alarm is triggered. This feature can optionally be used for program flow integrity protection, but it is not needed as the dual CPU and memory EDC mechanisms are far better suited to detect such attacks.

The Online Configuration Check (OCC) function controls the modification of relevant system settings. It is also useful as a measure against fault attacks and accidental changes. The content of the protected registers is permanently hashed and checked against a reference value. A violation generates an alarm event and leads to the secure state.

The TOE supports dynamical locking of dedicated peripherals by the SPAU. This way data flow between CPU and peripherals can be controlled. Manipulations utilizing access to specific peripherals can be prevented with this locking mechanism.

As physical effects or manipulative attacks may also target the program flow of the user software, a watchdog timer and a check point register are implemented. These features allow the user to check the correct processing time and the integrity of the program flow of the user software.

The covered security functional requirements are FPT_FLS.1, FDP_IFC.1, FPT_ITT.1, FDP_ITT.1 and FPT_PHP.3.

The HSL provides tearing safe write operations which can be utilized by the embedded software.

The covered security functional requirement is FPT_PHP.3.

The correct function of the TOE is only given in the specified range of environmental operating parameters. To prevent an attack exploiting that circumstance the TOE is equipped with a temperature sensor, glitch sensor and voltage sensor as well as backside light detection. The TOE falls into the defined secure state in case of a specified range violation. The defined secure state causes the chip internal reset process.

The covered security functional requirements are FRU_FLT.2 “Limited fault tolerance” and FPT_FLS.1 “Failure with preservation of secure state “.

7.4 SF_PLA: Protection against Logical Attacks

The TOE implements the Armv8-M Memory Protection Unit (MPU) with 8 regions and the Security Attribution Unit (SAU) with 8 regions according to [Armv8-M], ch. B10.

The SAU contains an Implementation Defined Attribution Unit (IDAU). The IDAU exempts the address ranges 4000 0000$_H$ - 5FFF FFFF$_H$ and A000 0000$_H$ - FFFF FFFF$_H$ from Security attribution.

During each start-up of the TOE the address ranges and MPU access rights are initialized by the Boot Software (BOS) with predefined values. The BOS maps a small region containing the start-up code for access of privilege software.

The SAU is disabled, and all addresses are marked secure and non-secure not callable.

The covered security functional requirements are FDP_ACC.2/AF, FDP_ACF.1/AF, FMT_MSA.1/AF/S, FMT_MSA.1/AF/NS, FMT_MSA.3/AF, FMT_SMF.1/AF and FMT_SMR.1/AF.

7.5 SF_HC: Hardware provided cryptography

The TOE is equipped with a random number generator as defined in the SFRs FCS_RNG.1/TRNG in chapter 6.1.1.

The covered security functional requirement is FCS_RNG.1/TRNG.

The TOE supports the encryption and decryption in accordance with the Advanced Encryption Standard (AES) and cryptographic key sizes of 128 bits or 192 bits or 256 bits that meet the standards as defined in chapter 6.1.2.

The TOE supports the encryption and decryption in accordance with the Triple TES (TDES) and cryptographic key sizes of 112 bits or 168 bits that meet the standards as defined in chapter 6.1.2.

The covered security functional requirement are FCS_COP.1/AES, FCS_COP.1/TDES and FCS_CKM.4.

7.6 SF_CS: Crypto Suite Services

The optional Crypto Suites utilizes the symmetric coprocessor and the asymmetric coprocessor of the hardware to implement standard cryptographic algorithms.

For details, see confidential Security Target.

8 Hash values of libraries

This chapter list the SHA256 hashes of the libraries from section 1.4.2.2

Table 63 SHA256 hash values

Lib	SHA256
NRG™ 06.10.0003	f16fc3955c07c279de59201c291356890e48a59e62f9b75230873e25bab0868c
NRG™ 06.10.0005	f5e97e04d2e2ee7cbe0f362f6c18e5e9e8db79d5ccecdd971ea07545be44efc6
UMSLC	74091c50254bc348a48a2e261a125735dca41f2419cd9541c3e6fbfdff63b529
HSL	b4b63cc2fc97a52ede25b1ceba4333f3d77576ce7ae42228c480454005a95436
Crypto Suite 4.06.002	CS-SLC26V19-sym-cipher-aes.lib: SHA256=b7dffd597478595a49f6b4d013ced083ebbdd3047ec5716def5af08092a30939 CS-SLC26V19-asym.lib: SHA256=66448342dfd4fac41576883bb50c25c821824cb4f752d1f838ffaa1272697656 CS-SLC26V19-asym-base.lib: SHA256=c84408aedcfcb065082fdbffc25f1e606d71a94850be3ce5dc65f147ad3c8b6c CS-SLC26V19-sym-cipher.lib: SHA256=2984b13b2b85a9556004c6b5dd06706079221c40c037cc3c8734286bd047d77b CS-SLC26V19-sym-mac-cmac.lib: SHA256=34ed79cf7046d17340240b4a44ed956f621f42c0d4db1ddcb819ddde4da27c78 CS-SLC26V19-sym-mac-cmac-aes.lib: SHA256=d9135e7aee35a936a41772c430433d3e6554b7c23f189b7471d303cbcc8e1eb8 CS-SLC26V19-core.lib: SHA256=50c92b0a71704d9005f488ae039d0d0713343465fea188817c1d0c20640f03ef CS-SLC26V19-sym-cipher-des.lib: SHA256=be9bd42d07003de93dbba556b39d9f279f578f0f1cad97fdc12e8c5217513843 CS-SLC26V19-asym-ecc.lib: SHA256=5554a35f131840780b11d796dfe8a194c4bda03292fc29c60a9bd8d74c400b6a CS-SLC26V19-sym-hash.lib: SHA256=abf3d371711623c2202d2bd0593007be505ce17f263c3f40d4a60387d76a3078 CS-SLC26V19-rng-hwrng.lib: SHA256=3cc9b9327d423e27425f2cd40dd60514054f03373090399f9cfd2098af30eb7a CS-SLC26V19-sym-mac.lib: SHA256=bfb66c47301735f0f8edab0e490fc405b6d6afd88fac514cf7c91a6ec15b093a CS-SLC26V19-sym-mac-retailmac.lib: SHA256=c3be4a9431d750a8f470ce7ccfd54db80ebbd9d10b89c3558910cff08f98501c CS-SLC26V19-rng.lib: SHA256=76bb07548a4eb32f517dd021727d95b8090268bf16d3fde67b55b4ff83ee25dd CS-SLC26V19-asym-rsa.lib: SHA256=bdc31248aeadc08795c354c3a42e052112def05fd627c95509ac3f01afc52a2b CS-SLC26V19-sym-hash-sha.lib:

Lib	SHA256
	SHA256=e5e603b8521f4b726ee63096bb4a1bc68b5263b544a061ebcb610f3db8abd012 CS-SLC26V19-sym.lib: SHA256=19eb1a0d2e713c5f5a73a55f6bf33f339c7fb274d9376b9508e46447865cd0e3 CS-SLC26V19-asym-tlbx.lib: SHA256=d37c33afed2570948161ce87e0c6ff498bb8d89b65d24159b92fe088094b3453
Crypto Suite 4.08.001	CS-SLC26V19-sym-cipher-aes.lib: SHA256=4dec4a3217d198987a194c12fccb93d929e62bdf1171d1f67167f2746bc847ac CS-SLC26V19-asym.lib: SHA256=8849cf18f4db4988900b58e44b26a363dfde13dff4cbb7104a905972ac233a24 CS-SLC26V19-asym-base.lib: SHA256=2e9cb44bed1c4c17c9e773a7a3664575188389bc3ab7b047c58e1d00bcaef94b CS-SLC26V19-sym-cipher.lib: SHA256=dbcc346bdc587b5816901d2977fd2f563a961a01b41ecdb79a08b0881cb82615 CS-SLC26V19-sym-mac-cmac.lib: SHA256=c6d34bb9bbcdfdbdff49f49b051783f33f55c30a0fabc896479e2d9ffe590fb3 CS-SLC26V19-sym-mac-cmac-aes.lib: SHA256=0cefd5e88d5b4f75938225d2eed0886c043bd513e2a4d22f942034b4f9a2ca1d CS-SLC26V19-core.lib: SHA256=4a0f807bbd91e3e9915175935cdac477de03ebb02c0e51d31a76ef2e9d405f73 CS-SLC26V19-sym-cipher-des.lib: SHA256=f937bbcc33da99458dd5433ca9a8dd04dcbda63b044045f41ece3e82ec506632 CS-SLC26V19-asym-ecc.lib: SHA256=304fdae95f16fec9ddaeb826134c6de795086bae185ee0d1eb8da52596dcb188 CS-SLC26V19-sym-hash.lib: SHA256=9922a09cf71c3a2c77cb092b7d6c57a6195759b152836be13fc82e604ce0ca8e CS-SLC26V19-rng-hwrng.lib: SHA256=253e6e8575e68985424b5ac5f9273d7b8a33fcb1aaa0c6ba830b3cec8b07a345 CS-SLC26V19-sym-mac.lib: SHA256=5ab7391e5b7f4d2cac6562932fe819a816d6c6afaad603aef48aec792b9afde2 CS-SLC26V19-sym-mac-retailmac.lib: SHA256=7644a2f3e04404b5fcc9fb01278f4f2d9202d7b803a2e550fa7b8eda20ac5f55 CS-SLC26V19-rng.lib: SHA256=ff47f05c2219e7bc65a694a16c9bcfe0645822ed983ef70b2e9fcdb2d9168607 CS-SLC26V19-asym-rsa.lib: SHA256=17ee5e6488b1084688536a6e689cca54192c8ad378fe3a6268a94a8745b608fd CS-SLC26V19-sym-hash-sha.lib: SHA256=ccf10dd583207276a527f68cf0f2a90c166c4f73d41955f41c25e637090550d1 CS-SLC26V19-sym.lib: SHA256=3e43477d7f6d9dccd1b046d382259d5257d28ac99ce69188cac4987960f11223 CS-SLC26V19-asym-tlbx.lib: SHA256=7a6dc4a48e012fad496257075d821abfe8ecc11c5ea8be55f993548f7d4091df

9 Cryptographic Table

The cryptographic table contains all cryptographic algorithms, which are either used by the dedicated software or are provided by the hardware or the Crypto Suites. Please note that the Crypto Suites provide more flexibility in combining cryptographic primitives than what is allowed in the referenced standard. To be compliant to the standard, the user of the Crypto Suite is responsible for using the correct combination of primitives.

Table 64 Cryptographic table

Purpose	Cryptographic operation	Key size in bits	Standards
Confidentiality	AES in ECB mode provided by hardware	128, 192, 256	[FIPS 197] [SP 800-38A]
Confidentiality	AES encryption and decryption in ECB, CBC mode	128, 192, 256	[FIPS 197] [SP 800-38A]
Integrity	AES CMAC mode	128, 192, 256	[FIPS 197] [SP 800-38B]
Confidentiality	TDES in ECB mode provided by hardware	112, 168	[SP 800-67] [SP 800-38A]
Confidentiality	TDES encryption and decryption in ECB, CBC mode	112, 168	[SP 800-67] [SP 800-38A]
Integrity	TDES Retail MAC mode	112	[SP 800-67] [ISO9797_1]
Confidentiality	RSAEP RSA encryption	1024 - 4224	[PKCS#1], ch. 5.1.1
Confidentiality	RSADP RSA decryption with exponential representation	1024 -2112	[PKCS#1], ch. 5.1.2, 2a
Confidentiality	RSADP RSA decryption with CRT representation	1024 -4224	[PKCS#1], ch. 5.1.2, 2b
Integrity	RSA signature generation RSASP1	1024 -4224	[PKCS#1], ch. 5.2.1, 2b
Integrity	RSA signature verification RSAVP1	1024 -4224	[PKCS#1], ch. 5.2.2
Integrity	ECDSA signature generation	160-521	[FIPS 186-4], 6.4 [FIPS 186-5], 6.4.1
Integrity	ECDSA signature verifications	160-521	[FIPS 186-4], 6.4 [FIPS 186-5], 6.4.2
Key agreement	ECDH key agreement	160-521	[SP 800-56A], ch. 5.7.1.2
Authenticated encryption	AES CCM[1]	128	[SP 800-38C]
Key agreement	Finite Field Diffie-Hellman	1024-2048	[PKCS#3], ch. 7.2
Key derivation	KDF in counter mode with AES CMAC as PRF[1]	128	[SP 800-108], 5.1 [SP 800-38B], 6.2
Hash	SHA-1 Hash digest	N/A	[FIPS 180-4]
Hash	SHA-2 Hash digest 224, 256, 384, 512, 512/224, 512/256 bits	N/A	[FIPS 180-4]

[1] This algorithm is only used by the Flash Loader and not provided as service for the Embedded Software

Purpose	Cryptographic operation	Key size in bits	Standards
Random	Physical RNG PTG.2	N/A	[AIS 31]
Random	Hybrid physical RNG PTG.3	N/A	[AIS 31]

Acronyms

Acronym	Description
AES	Advanced Encryption Standard
AFM	Advanced Framing Mode
CL	ContactLess
CSP	Chip Scale Package
DES	Data Encryption Standard
ECC	Elliptic Curve Cryptography or Error Correction Code
EDC	Error Detection Code
FFC	Finite Field Cryptography
HBR	High Baud Rate
ISS	Instruction Stream Signature
MCICE	Memory Cache Integrity Confidentiality Engine
MPU	Memory Protection Unit
NVIC	Nested Vectored Interrupt Controller
NVM	Non-Volatile Memory
OCC	Online Configuration Check
RAM	Random Access Memory
RF	Radio Frequency
RNG	Random Number Generator
ROM	Read Only Memory
RSA	Rivest Shamir Adleman
SAU	Security Attribution Unit
SDM	Software Debug Module
SE	Security Extension
SPI	Serial Peripheral Interface
SPM	Security Policy Model
SWP	Single Wire Protocol
TDES	Triple DES
TOE	Target Of Evaluation
UMSLC	User Mode Security Life Control
VHBR	Very High Baud Rate

References

[AIS 31]	Anwendungshinweise und Interpretationen zum Schema AIS 31, Version 3, 15.05.2013 Bundesamt für Sicherheit in der Informationstechnik
[AIS 46]	Anwendungshinweise und Interpretationen zum Schema AIS 46, Version 3, 2013-12-04 Bundesamt für Sicherheit in der Informationstechnik
[ANSSI]	ANSSI: "Avis relatif aux paramètres de courbes elliptiques définis par l'Etat français" in: Journal Officiel de la République Française (JORF)
[Armv8-M]	Arm® v8-M Architecture Reference Manual, ARM part number: AR100-DA-78000-r0p1-10eac0
[CC1]	Common Criteria for Information Technology Security Evaluation Part12: Security Functional Requirements; Version 3.1 Revision 5 April 2017, CCMB-2017-04-001
[CC2]	Common Criteria for Information Technology Security Evaluation Part 2: Security Functional Requirements; Version 3.1 Revision 5 April 2017, CCMB-2017-04-002
[CC3]	Common Criteria for Information Technology Security Evaluation Part 3: Security Assurance Requirements; Version 3.1 Revision 5 April 2017, CCMB-2017-04-003
[FIPS 180-4]	NIST: FIPS publication 180-4: Secure Hash Standard (SHS), August 2015
[FIPS 186-4]	NIST: FIPS publication 186-4: Digital Signature Standard (DSS), July 2013
[FIPS 186-5]	NIST: FIPS publication 186-5: Digital Signature Standard (DSS), February 3, 2023
[FIPS 197]	NIST: FIPS publication 197: Advanced Encryption Standard (AES), May 9, 2023
[ICAO]	ICAO Doc 9303, Machine Readable Travel Document, 8th edition, 2021, Part 11: Security Mechanisms for MRTDs
[ISO9798_2]	ISO/IEC 9798-2: 2008 - Information Technology - Security techniques - Entity authentication - Part 2: Mechanisms using authenticated encryption. Fourth edition 2019-06
[ISO9797_1]	ISO/IEC 9797-1:2011 Information technology — Security techniques — Message Authentication Codes (MACs) — Part 1: Mechanisms using a block cipher
[ISO15946]	ISO/IEC 15946-5:2022
[JIL]	Joint Interpretation Library, Application of Attack Potential to Smartcards, Version 3.2 November 2022
[PKCS#3]	Diffie-Hellman Key-Agreement Standard An RSA Laboratories Technical Note Version 1.4 Revised November 1, 1993
[PKCS#1]	PKCS #1: RSA Cryptography Standard, v2.2, October 27, 2012, RSA Laboratories

[PP0084]	Security IC Platform Protection Profile with Augmentation Packages, Version 1.0, 13.01.2014, BSI-CC-PP-0084-2014
[RFC 5639]	IETF: RFC 5639, Elliptic Curve Cryptography (ECC) Brainpool Standard Curves and Curve Generation, March 2010, http://www.ietf.org/rfc/rfc5639.txt
[RFC 7748]	Internet Research Task Force (IRTF), Request for Comments: 7748, January 2016
[RFC 8928]	Internet Research Task Force (IRTF), Request for Comments: 8928, November 2020
[SEC2]	SEC 2: Recommended Elliptic Curve Domain Parameters Certicom Research, January 27, 2010, Version 2.0
[SP 800-22]	National Institute of Standards and Technology(NIST), Technology Administration, US Department of Commerce, Special Publication 800-22, A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications, in the revision 1a as of April 2010
[SP 800-38A]	National Institute of Standards and Technology (NIST), Technology Administration, U.S. Department of Data Encryption Standard, NIST Special Publication 800-38A, Edition 2001
[SP 800-38B]	National Institute of Standards and Technology (NIST), Special Publication 800-38B, May 2005
[SP 800-38C]	NIST SP 800-38C: Recommendation for Block Cipher Modes of Operation: The CCM Mode for Authentication and Confidentiality, 2004-05 (up-dated: 2007-07-20)
[SP 800-56A]	National Institute of Standards and Technology (NIST), Special Publication 800-56A, Revision 3, April 2018
[SP 800-108]	National Institute of Standards and Technology (NIST), Recommendation for Key Derivation Using Pseudorandom Functions, NIST SP 800-108r1, August 2022
[TACHO]	Appendix 11 of Annex I(B) of Commission Regulation (EEC) No. 1360/2002 [5] - Common Security Mechanisms
[SP 800-67]	National Institute of Standards and Technology (NIST), Special Publication 800-67, Revision 2, November 2017

Revision history

Version	Date	Description
Rev. 1.2	2024-08-26	Predecessor ST
Rev. 1.7	2025-07-09	Release