NXP Secure Smart Card Controller N7121 with IC Dedicated Software and Crypto Library

Security Target Lite

Rev. 1.1 — 31 May 2019

Evaluation document

BSI-DSZ-CC-1040

Document information

Information	Content
Keywords	Common Criteria, Security Target, Security IC, N7121
Abstract	This document is the Security Target of the NXP Smart Card Controller N7121 with IC Dedicated Software. The document describes the security Functionality provided by the IC and its software.

Revision history

Table 1. Revision history

Version	Release date	Change notice
1.0	2018-11-30	Initial version based on full Security Target v1.4
1.1	2019-05-31	Updated version based on full Security Target v1.5

1 Introduction

1.1 ST reference

Security Target Lite NXP Secure Smart Card Controller N7121 with IC Dedicated Software and Crypto Library, Version 1.1, NXP Semiconductors Germany GmbH, 31 May 2019.

1.2 TOE reference

The TOE is named NXP Secure Smart Card Controller N7121 with IC Dedicated Software and Crypto Library. In this document, the TOE is abbreviated to N7121. All components of the TOE and their respective version numbers are listed in Section 1.4.1.

1.3 TOE overview

The TOE is the hard macro NXP Secure Smart Card Controller N7121 with IC Dedicated Software and Crypto Library, or in short N7121, which comprises hardware, software (security IC Dedicated Software), and documentation. The N7121 is self-sufficient at the boundary of the hard macro and can be instantiated within packaged products. The TOE does not include a customer-specific Security IC Embedded Software, however, it provides secure mechanisms for customers to download and execute their code on the TOE.

1.3.1 Hardware

The IC hardware is a microcontroller incorporating a central processing unit (CPU), memories accessible via a Memory Management Unit (MMU), cryptographic coprocessors, other security components, contact-based and contactless communication interfaces as well as a general purpose I/O interface which can be used to directly use peripherals of the TOE such as the cryptographic coprocessors. The central processing unit supports a 32-/16-bit instruction set optimized for smart card applications. On-chip memories are ROM, RAM and Flash. The Flash can be used as data or program memory. It consists of highly reliable memory cells, which are designed to provide data integrity. The Flash memory is optimized for applications that require reliable non-volatile data storage for data and program code. Dedicated security functionality protects the contents of all memories. The logical Flash size can be configured in 1kB steps. The IC integrates coprocessors for AES, DES (both within the new Crypto2+ coprocessor) and a new 128 bit Public Key Crypto Coprocessor (Fame3) to support the implementation of asymmetric cryptographic algorithms.

Note: *Please note that the Flash memory is also referred to as Non-Volatile Memory (NVM) in this Security Target.*

The IC Embdedded Software can either be located in ROM or Flash, see Table 3.

1.3.2 Software

The IC Dedicated Software comprises IC Dedicated Test Software for test purposes and IC Dedicated Support Software. The IC Dedicated Support Software consists of the Boot Software, which controls the boot process of the hardware platform. Furthermore, it provides a Firmware Interface and optionally a Library Interface, simplifying access to the hardware for the Security IC Embedded Software. The IC Dedicated Support Software also comprises optional software components, i.e.,

- two logical cards (A and B),
- a System Mode OS which offers ready-to-use resource and access management for customer applications that do not want to be exposed to the more low-level features of the TOE,
- the System Mode OS also provides a Secure User Mode Box, which further restricts the access of code executed in User Mode (UM),
- a Flash Loader OS which supports download of code and data to Flash by the Composite Product Manufacturer before Operational Usage (e.g. during development), and
- a crypto library which provides simplified access to frequently used cryptographic algorithms AES, TDES, RNG, RSA, ECC, hashing and Utilities.

The availability of these software components depends on the different TOE configurations defined in Section 1.4.1.

1.3.3 Documentation

The documentation includes a Product Data Sheet with several addenda, an Instruction Set Manual, a Guidance and Operation Manual, User Manuals for cryptographic functions and Utilities as well as a Wafer and Delivery Specification. This documentation comprises a description of the architecture, the secure configuration and usage of the IC hardware platform and the IC Dedicated Support Software by the Security IC Embedded Software. As some parts of the IC Dedicated Support Software are optional, the respective documentation is optional as well and depends on the TOE configurations chosen by the customer. The dependencies and list of documentation is given in Table 3.

1.3.4 Usage and major security functionality of the TOE

The security functionality of the TOE is designed to act as an integral part of a complete security system in order to strengthen the design as a whole. Several security mechanisms are completely implemented in and controlled by the TOE. Other security mechanisms allow for configuration by or even require support of the Security IC Embedded Software.

N7121 provides high security for smartcard applications and in particular for being used in the banking and finance market, in electronic commerce, or in governmental applications. Hence, the N7121 shall maintain

- the integrity and the confidentiality of code and data stored in its memories,
- the different TOE modes with the related capabilities for configuration and memory access, and
- the integrity, the correct operation and the confidentiality of security functionality provided by the TOE.

This is ensured by the construction of the TOE and its security functionality.

The N7121 basically provides a hardware platform and crypto library for an implementation of a smart card application with

- functionality to calculate Data Encryption Standard (Triple-DES) with up to three keys,
- functionality to calculate Advanced Encryption Standard (AES) with different key lengths,
- functionality to calculate RSA, RSA key generation, RSA public key computation,

- functionality to calculate ECDSA (ECC over GF(p)) signature generation and verification, ECDSA (ECC over GF(p)) key generation, ECDH (ECC Diffie-Hellmann) key-exchange, and full point addition(ECC over GF(p)) over any Weierstrass curves from size 128 bits to size 640 bits with co-factor equal 1,
- basic support of the PACE protocol ([TR-03110-1], [TR-03110-2], [TR-03110-3], [TR-03110-4]) as ECC base-point operations are protected against leakage and fault injection,
- KeyStore feature for secure key management,
- secure copy, move, and compare operations provided by the crypto library,
- functionality to compute SHA-1, SHA-224, SHA-256, SHA-384, and SHA-512,
- a True Random Number Generator,
- a Hybrid Deterministic Random Number Generator,
- a Hybrid Physical Random Number Generator,
- a Physical Unclonable Function (PUF),
- memory management control and memory encryption,
- physical protection via sensors on the chip and chip shielding, and
- two completely separated logical cards A and B, each with System Mode and User Mode.

Further functionality of the TOE which **does not correspond to security functionality** as defined in this Security Target is

- ISO/IEC 7816 contact interface with UART and ISO/IEC 14443A contactless interface,
- a general purpose communication interface which can be used to directly access peripherals of the TOE,
- an Undocumented Function (UDF), i.e., a proprietary operation used for data blinding, and
- cyclic redundancy check (CRC) calculation.
- KoreanSeed Library, providing cryptographic operations using the 128-bit block chiper SEED.

1.3.5 TOE type

The TOE NXP Secure Smart Card Controller N7121 with IC Dedicated Software and Crypto Library, or in short N7121, is provided as IC hardware platform with IC Dedicated Software for various operating systems and applications with high security requirements.

1.3.6 Required non-TOE hardware/software/firmware

Besides the conventional communication interfaces, the TOE provides a general purpose I/O interface. It is not required to use this interface, however, it can be used to access major security features of the TOE. This certification does not address the communication via the general purpose I/O interface, however, the TOE implements countermeasures against misuse.

1.4 TOE description

1.4.1 Evaluated configurations and TOE components

The TOE features different types of memories, some are configurable to the customer others are fixed, as shown in the following table.

Table 2. Memories of the TOE

Memory type	Memory size	Description
NVM	configurable up to 342 KBytes	The size of the Non-Volatile Memory.
ROM	Configurable to 0 KBytes or 150 KBytes	Size of the Read-Only Memory.
RAM	12 kBytes	Size of the Random-Access Memory. Size available to customer depends on ordered configuration (e.g., availability of MIFARE).

The TOE provides different configuration options a customer can make either in the ordering process (Ordering configurations) or after the delivery (Post-Delivery Configurations). The following table lists configurations with impact on the security functionality.

Default values are indicated as **bold text**.

Table 3. TOE configuration options

Product option	Choices	Description
Ordering configurations		
NVM Size	configurable in 1kB steps up to 342 KBytes	The Flash memory size is logically configurable, within the given step size. ***Note:*** *If utilized, the Flashloader occupies 16 kB of storage, which are freed up after its usage.*
Customer Type	• System Mode customer • **User Mode customer**	Depending on this choice, the customer has access to the System Mode of the logical cards (System Mode customer) or not (User Mode customer). In the first case, customers can store the Security IC Embedded Software in the System Mode of the available logical cards. Otherwise, the NXP System Mode OS is in place on each available logical card in System Mode, while the customer can only access the less privileged User Mode.
Use Flash Loader	• **Yes** • No	Depending on this choice, the TOE provides the functionality of a Flash Loader such that customers can load their code to the NVM memory. If the Flash Loader is available, the Library Interface and the N7121 Crypto Library become mandatory. If the Flash Loader is not available, the customer can still decide whether the Security IC Embedded Software will be stored in ROM or Flash during the development process.

Beside these configurations, further Ordering and Post-Delivery Configurations are possible. However, these do not affect the Security Functionality defined in this ST. These configurations are listed below to provide a full picture:

Ordering Configurations without security impact:

- User ID settings,
- different options for contact-based and contactless communication,
- available data rates (106kbit/s, 106-848kbit/s, 106-848kbit/s and VHBR rates up to 3.2Mbit/s, or all),
- ATS/ATR check during testing,

- Enable or disable Chip Health Mode (CHM).

The TOE does not provide any functionality to tailor its available security functionality after delivery. Via Post-Delivery Configurations (PDC), the following configurations are possible which have no impact on the security functionality provided by the TOE:

- MIFARE DESFire EV2 (2K, 4K, 8K, 16K, 24K, 32K, or disabled) (resides in UM of card A) and
- MIFARE PLUS2 (2K, 4K, or disabled) (resides in UM of card A).

Note:

Logical card A provides MIFARE DESFire and/or MIFARE PLUS2 in its User Mode. The NXP SM OS is mandatory on Card A and cannot be replaced. It configures the Menory Management Unit in such a way that software running in User Mode of Card A cannot interfere with other software running on the TOE and related memories on the TOE if not explicitly allowed. The MIFARE Software does not provide any security functionality defined in this ST. This functionality is called "NXP Secure User Mode Box".

Note:

The CHM can be used for chip identification, application of post-delivery configurations, and functional self-tests of the TOE. If the CHM is not available, chip identification and post-delivery configurations is still available, however, a customer has to spend more effort to access the respective functionality. The functional self-tests which are available in the CHM are not part of the Security Functionality defined in this ST.

Depending on the customer choices, the N7121 comprises the following deliverables:

Table 4. TOE deliverables

Type	Name	Release	Form of delivery
TOE components for all configurations			
IC Hardware	N7121	B1	Hard macro instantiated within a wafer, modules and package.
IC Dedicated Test Software	Test Software	9.2.3	On-chip software
IC Dedicated Support Software	Boot Software	9.2.3	On-chip software
	Firmware	9.2.3	On-chip software
Document	*NXP Secure Smart Card Controller N7121 – Overview*, Product data sheet [DSheet]	3.2	Electronic document (PDF via NXP DocStore)
Document	*NXP Secure Smart Card Controller N7121 – Instruction Set Manual*, Objective data sheet addendum [DSheet_InSet]	3.0	Electronic document (PDF via NXP DocStore)
Document	*NXP Secure Smart Card Controller N7121 – Chip Health Mode*, Objective data sheet addendum [DSheet_CHM]	3.0	Electronic document (PDF via NXP DocStore)
Document	*NXP Secure Smart Card Controller N7121 – Peripheral Configuration and Use*, Objective data sheet addendum [DSheet_Periph]	3.1	Electronic document (PDF via NXP DocStore)

Type	Name	Release	Form of delivery
Document	*NXP Secure Smart Card Controller N7121 – MMU Configuration and NXP Firmware Interface Specification*, Objective data sheet addendum [DSheet_MMU]	3.2	Electronic document (PDF via NXP DocStore)
Document	*NXP Secure Smart Card Controller N7121, Information on Guidance and Operation*, Guidance and operation manual [GOM]	3.2	Electronic document (PDF via NXP DocStore)
Deliverables of the Flash Loader OS			
IC Dedicated Support Software	Flashloader OS	1.2.5	On-chip software
Document	*NXP Secure Smart Card Controller N7121 – Flashloader OS*, Objective data sheet addendum [DSheet_FL]	3.0	Electronic document (PDF via NXP DocStore)
Deliverables of the Library Interface			
IC Dedicated Support Software	Library Interface	9.2.3	On-chip software
Library	Communication Library	6.0.0	Electronic files (object files via NXP DocStore)
Library	CRC Library	1.1.8	Electronic files (object files via NXP DocStore)
Library	Memory Library	1.2.3	Electronic files (object files via NXP DocStore)
Library	Flash Loader Library	3.6.0	Electronic files (object files via NXP DocStore)
Document	*NXP Secure Smart Card Controller N7121 – Shared OS Libraries*, Objective data sheet addendum [DSheet_LibInt]	3.0	Electronic document (PDF via NXP DocStore)
Deliverables of the System Mode OS (for UM customers)			
IC Dedicated Support Software	System Mode OS	13.2.3	On-chip software
Document	*NXP Secure Smart Card Controller N7121 – NXP System Mode OS*, Objective data sheet addendum [DSheet_SMOS]	3.2	Electronic document (PDF via NXP DocStore)
Deliverables of the crypto library			
IC Dedicated Support Software	Crypto Library	0.7.6	On-chip software
Package Random Number Generation			
Library	RNG Lib	0.7.6	Electronic files (object files via NXP DocStore)

Type	Name	Release	Form of delivery
Library	RNG HealthTest Lib	0.7.6	Electronic files (object files via NXP DocStore)
Document	*N7121 Crypto Library – RNG Library*, Preliminary user manual [UM_RNG]	1.2	Electronic document (PDF via NXP DocStore)
Package Symmetric Ciphers			
Library	Sym. Cipher Lib	0.7.6	Electronic files (object files via NXP DocStore)
Document	*N7121 Crypto Library – Symmetric Cipher Library (SymCfg)*, Preliminary user manual [UM_SymCfg]	1.4	Electronic document (PDF via NXP DocStore)
Package KeyStore			
Library	KeyStoreMgr Lib	0.7.6	Electronic files (object files via NXP DocStore)
Document	*N7121 Crypto Library – KeyStoreMgr Library*, Preliminary user manual [UM_KeyStore]	1.1	Electronic document (PDF via NXP DocStore)
TOE components required for the packages Random Number Generation and Symmetric Ciphers			
Library	Sym. Utilities Lib	0.7.6	Electronic files (object files via NXP DocStore)
Document	*N7121 Crypto Library – Utils Library*, Preliminary user manual [UM_SymUtils]	1.1	Electronic document (PDF via NXP DocStore)
Package RSA Encryption / Decryption			
Library	RSA Lib	0.7.6	Electronic files (object files via NXP DocStore)
Document	*N7121 Crypto Library – RSA Library*, Preliminary user manual [UM_RSA]	1.4	Electronic document (PDF via NXP DocStore)
Package RSA Key Generation			
Library file	RSA Key Generation Lib	0.7.6	Electronic files (object files via NXP DocStore)
Document	*N7121 Crypto Library – RSA Key Generation Library*, Preliminary user manual [UM_RSAKeyGen]	1.3	Electronic document (PDF via NXP DocStore)
Package ECC over GF(p)			
Library	ECC Lib	0.7.6	Electronic files (object files via NXP DocStore)

Type	Name	Release	Form of delivery
Document	*N7121 Crypto Library – ECC over GF(p) Library*,Preliminary user manual [UM_ ECC]	2.1	Electronic document (PDF via NXP DocStore)
Package SHA			
Library	SHA Library & Hash Library	0.7.6	Electronic files (object files via NXP DocStore)
Document	*N7121 Crypto Library – SHA Library*, Preliminary user manual [UM_SHA]	1.1	Electronic document (PDF via NXP DocStore)
Document	*N7121 Crypto Library – HASH Library*, Preliminary user manual [UM_HASH]	1.2	Electronic document (PDF via NXP DocStore)
TOE components required for the packages RSA Encryption / Decryption, RSA Key Generation, ECC over GF(p), and SHA			
Library	Asym. Utilities Lib	0.7.6	Electronic files (object files via NXP DocStore)
Document	*N7121 Crypto Library – UtilsAsym Library*, Preliminary user manual [UM_AsymUtils]	1.3	Electronic document (PDF via NXP DocStore)
TOE components required for all packages			
Document	*N7121 Crypto Library, Information on Guidance and Operation*, Product user manual [GOM_CL]	3.0	Electronic document (PDF via NXP DocStore)

1.4.2 Physical scope of the TOE

The N7121 is manufactured in 40 nm CMOS technology. A block diagram of the IC hardware is depicted in Figure 1.

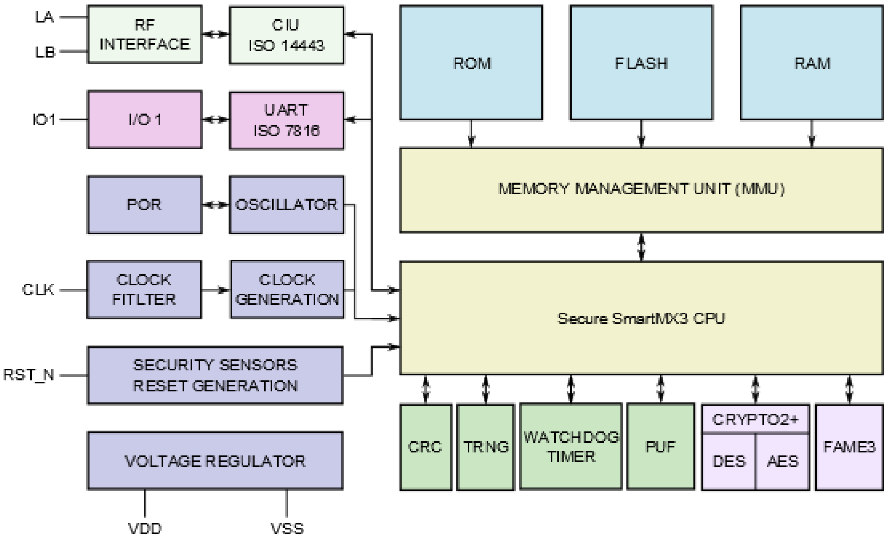

Figure 1. Block diagram of the TOE

The N7121 consists of the IC hardware and IC Dedicated Software. The IC Dedicated Software is composed of IC Dedicated Test Software for test purposes and IC Dedicated Support Software. The IC Dedicated Support Software contains the Boot Software, the Firmware Interface, the Library Interface, the cryptographic libraries, the System Mode OS and the Flash Loader OS. All other software is called Security IC Embedded Software. The Security IC Embedded Software is not part of the TOE (Application Note 2 of [PP]).

Please note that not all parts of the IC are defined as TOE. In addition to the conventional contact-based and contactless communication interfaces, the TOE provides a general purpose I/O interface which is directly connected to the internal SFR bus. This interface can be connected to an I2C interface for instance which is not part of the evaluation. The Security Functionality of the TOE does not rely on the communication interface connected to this interface. However, the TOE implements countermeasures against misuse.

1.4.3 Logical scope of the TOE

1.4.3.1 Hardware description

The TOE distinguishes different TOE modes as depicted in the following figure:

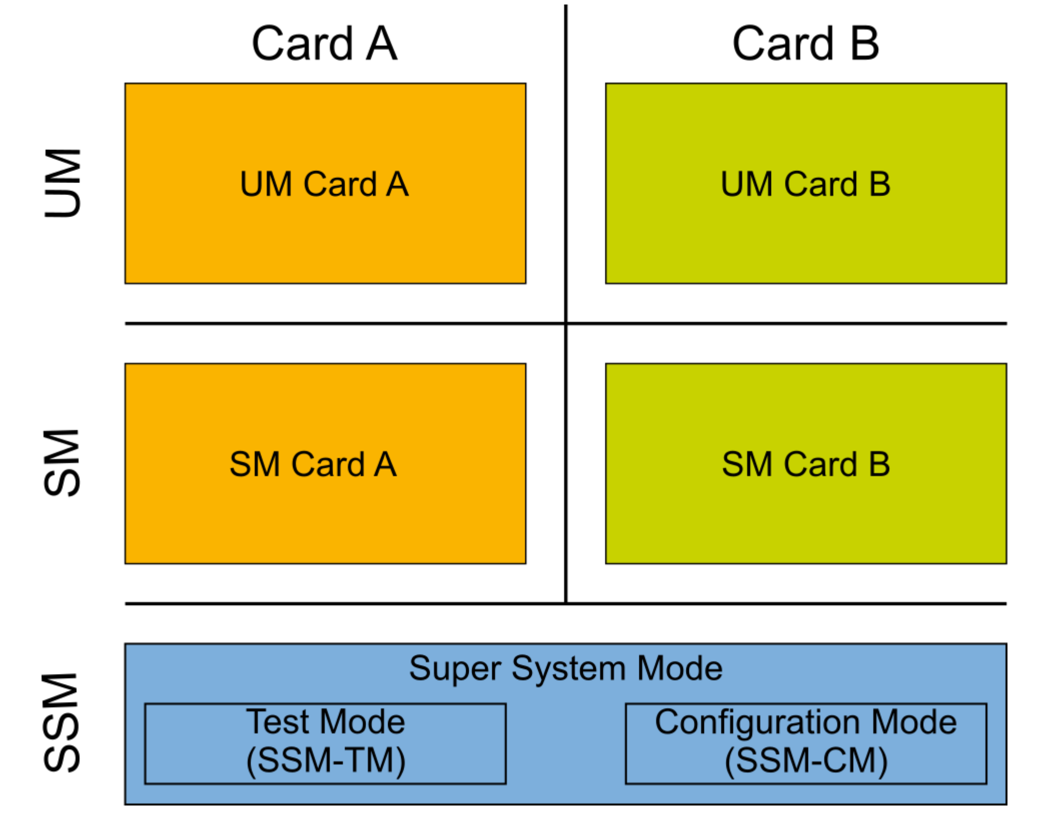

Figure 2. TOE modes overview

The Super-System Mode (SSM) is not available to the Security IC Embedded Software. It executes the Boot Software and Firmware.

The Test Mode (SSM-TM, short TM) and Configuration Mode (SSM-CM, short CM) have extended access rights compared to the Super-System Mode. The CPU however does not distinguish between SSM, TM, and CM. It only distinguishes between SSM, SM, and UM. In TM, the IC Dedicated Test Software is executed and is blocked before delivery. It includes the test operating system, test routines for the various blocks of the circuitry, control flags for the status of the Flash's manufacturer area and shutdown functions to ensure that security relevant test routing cannot be executed after Phase 3 of the life cycle defined in [PP]. Moreover, the IC Dedicated Test Software is used by NXP to download code related to System Mode or User Mode. A customer has no access to the IC Dedicated Test Software. The Configuration Mode is used to configure the TOE in the boot phase and to apply Post-Delivery Configurations.

The N7121 is able to control two different logical phases. After production of the Security IC, every start-up or reset completes with execution of the IC Dedicated Test Software. The test functionality is disabled at the end of the production test. Afterwards, every start-up or reset ends up in System Mode or User Mode, depending on the configuration ’Customer Type’ selected by the customer.

The TOE further provides the System Mode (SM) and User Mode (UM) which are available for the IC Embedded Software. If both logical cards A and B are available, each card implements its own SM and UM, which are completely separated from each other. The NXP System Mode OS applies a pre-configuration of the MMU to guarantee this separation. If the NXP System Mode OS is not available (SM customer) the MMU has to be configured by the System Mode User, i.e., via the Security IC Embedded Software. In case Logical Card A is not available, the TOE only distinguishes between UM, SM, and SSM (which includes TM and CM).The differentiation between both logical cards is only done via the MMU, the CPU only distinguishes between the different modes.

The System Mode has broader access to the hardware components available to the Security IC Embedded Software. The User Mode has restricted access to the CPU, specific Special Function Registers and the memories depending on the access rights granted by software running in System Mode. Please note that most Special Function Registers are implemented as RAM-based segment descriptors, initialized during start-up and controlled by the Memory Management Unit (MMU). The hardware components are controlled by the Security IC Embedded Software via Special Function Registers. Both are interrelated to the activities of the CPU, the Memory Management Unit, interrupt control, I/O configuration, NVM, timers and the coprocessors. A more detailed description of the Software available on and for the TOE is given in Section 1.4.3.2 .

The N7121 provides interrupts. Interrupts force a jump to a specific fixed vector address in the ROM or Flash. Any interrupt can therefore be controlled and guided by a specific part of the Security IC Embedded Software. In addition, the TOE provides user calls and system calls. These calls have to be explicitly done by the Security IC Embedded Software via dedicated CPU instructions. A user call starts the execution of related code dedicated to one lower privileged mode (Super System Mode to System Mode or System Mode to User Mode). A system call starts the execution of related code dedicated to one higher privileged mode (User Mode to System Mode or System Mode to Super System Mode).

The Watchdog timer is intended to abort irregular program executions by a time-out mechanism and is enabled and configured by the Security IC Embedded Software.

The TOE incorporates Flash, RAM, and program memory available in ROM. Access control to all three memory types is enforced by a Memory Management Unit (MMU). The System Mode OS provides a simplification of the resource management (e.g. MMU firewall settings, dynamic segment setup, and peripheral access control). The MMU partitions each memory into several parts, defined as objects in the Access Control Policy (see Section 6.1.8).

The Triple-DES coprocessor supports single DES and Triple-DES operations. Only Triple-DES is in the scope of this evaluation, in 2- key or 3-key operation with two/three 56-bit keys (112-/168-bit). The AES coprocessor supports AES operation with three different key lengths of 128, 192 or 256 bit. Both utilize the new Crypto2+ coprocessor. The physical random number generator provides true random numbers without pseudo random calculation. The new 128 bit Public Key Crypto Coprocessor (Fame3) supplies basic arithmetic functions to support the implementation of asymmetric cryptography, utilized by the asymmetric cryptographic library.

The TOE provides power saving modes with reduced activity. These are named IDLE Mode and SLEEP Mode, of which the latter one includes CLOCK STOP Mode.

The TOE protects secret data, which are stored on and operated by the TOE, against physical tampering. A memory encryption is added to the memories RAM, ROM and Flash such that data stored to these memories is encrypted. Chip shielding is added in form of active active shield. Light sensors are distributed over the chip area. Furthermore, the TOE is protected by voltage, temperature and frequency sensors. The security functionality of the IC hardware platform is mainly provided by the TOE, and completed by the Security IC Embedded Software. This causes dependencies between the security functionality of the TOE and the security functionality provided by the Security IC Embedded Software.

1.4.3.2 Software description

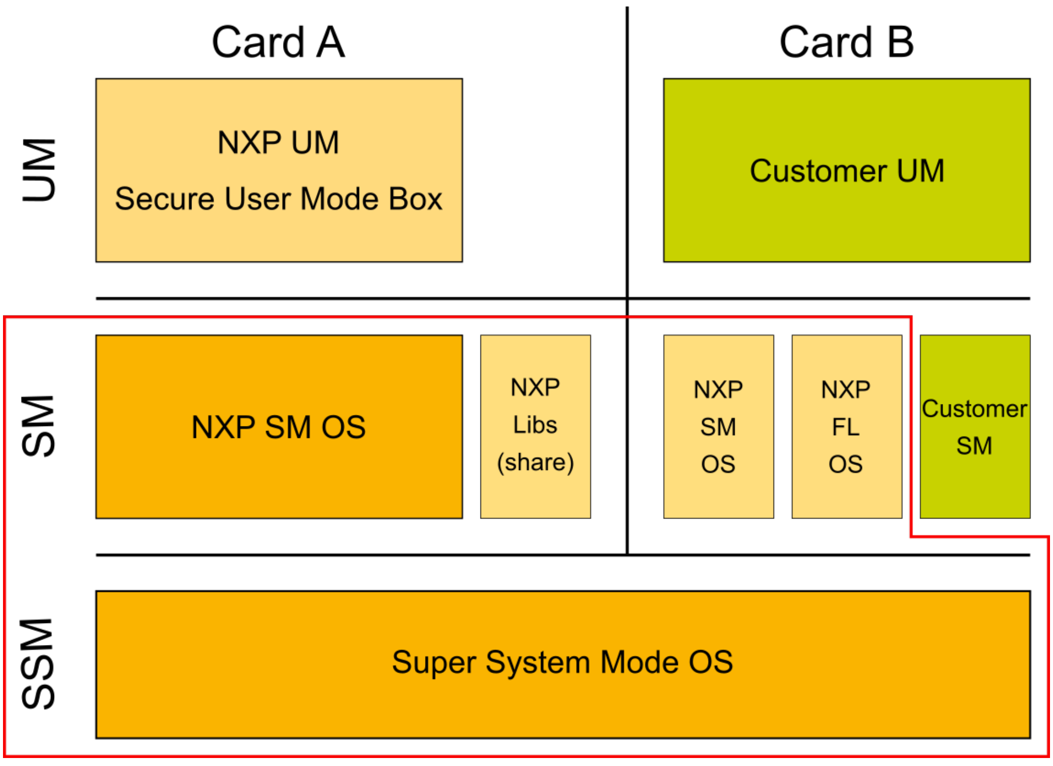

Figure 3. Software components of the TOE

Figure 3 shows the different pieces of the available software on and for the TOE. The scope of the TSF is additionally highlighted by a red box. The TOE supports two logical cards (Card A and Card B). Both logical cards are divided into a User Mode and a System Mode. Operating system and applications of a Security IC are developed by the customers and included under the heading Security IC Embedded Software. The logical location of the Security IC Embedded Software depends on the usage of the IC hardware platform. For User Mode customers, it is stored in the memories which belong to the User Mode of Card B. For System Mode customers, the Security IC Embedded Software can also be stored in memories which belong to the System Mode of Card B. Logical Card A is available for NXP code only. If this logical card is available, the SM of card A contains the NXP SM OS while User Mode of Card A contains NXP code like MIFARE Plus or MIFARE Desfire.

The separation between the two logical cards (Card A and Card B) is provided by the so-called ”Vertical IP firewall” which allows for having two completely separated logical cards on the same hardware without any unintended impact on each other. Because a logical card is also divided into a User Mode and a System Mode, it is possible to offer a security feature called ”Secure User Mode Box”.

This feature is of special importance as it allows for the integration of data and code to a certified product without any security impact. The Secure User Mode Box restricts the access rights for code running in UM of Card A, such that it has no influence on other modes and cards. For the "Secure User Mode Box", a fixed set of access rights are NXP-defined during production. The NXP System Mode OS does not provide any interfaces to the UM to change these access rights.

For logical card B, a "Secure User Mode Box" can be implemented as well. However, in this case the configuration of the User-Mode access rights is up to the customer and therefore out of the scope of this certification. This configuration is done as part of the ordering process.

Using shared memory segments it is possible to share data or code between the separated logical cards. The owner of a memory block has to explicitly allow this kind of sharing. The libraries are shared between the logical cards using this mechanism, reducing the footprint, as code only has to be present on the device once. An inter-card communication mechanism allows the currently active card to send a message to the inactive card with a request for card switching. This mechanism allows for the handover of execution between the logical cards.

The IC Dedicated Test Software is developed by NXP. It includes the test operating system, test routines for the various blocks of the circuitry, control flags for the status of the Flash's manufacturer area and shutdown functions to ensure that security relevant test routines cannot be executed illegally after Phase 3. The IC Dedicated Test Software is stored in ROM memory segments which belong to the Super-System Mode (SSM-TM).

The IC Dedicated Support Software comprises the following parts:

1. The Boot Software is executed after each reset of the TOE, i.e. every time when the TOE starts. It sets up the TOE and does some basic configuration of the hardware based on the settings stored in memories assigned to the SSM.
   The Boot Software is stored in ROM memories assigned to the SSM.
2. The Firmware Interface is stored in memories assigned to the SSM. It provides low-level flash management, the Post-Delivery Configuration feature and basic system functionality like self-testing, error-counter handling, PUF and reset functionality. Notice, that Boot Software and IC Dedicated Test Software also access the Firmware Interface. The 'one-time executed' part of the Firmware Interface (for instance PDC) is located in FLASH, the remaining parts are located in ROM.
3. The NXP System Mode OS is an Operating System developed by NXP. In general, the NXP System Mode OS provides a ready-to-use resource and access management for any customer application and does not expose the more low-level features, such as MMU configuration. It provides the feature of the Secure User-Mode Box.
   The NXP System Mode OS is a mandatory component of Card A, implementing its System Mode OS. It is responsible for sharing the different TOE libraries available. For System Mode customers who do not need any TOE library, the Flash Loader, or any NXP application running in User Mode of Card A, the System Mode OS on Card A is deactivated and cannot be executed. This results in the deactivation of Card A. For Card B the NXP System Mode OS is an optional component. However, it becomes mandatory for User Mode customers.
   The NXP System Mode OS can be stored in ROM or FLASH.
4. The Library Interface is an optional module and can be stored in any Card and mode. It provides simplified communication, CRC and memory functions to the Security IC Embedded Software. The Library Interface is required by the Flashloader OS and the Crypto Libraries.
5. The crypto library is an optional library which provides extended functionality and access to the following functionality to the Security IC Embedded Software:
   - Package Symmetric Ciphers for AES and TDES in various modes.
   - Package Random Number Generation which implements the hybrid deterministic RNG and hybrid physical RNG including health tests.

- Package RSA Encryption / Decryption: RSA encryption, decryption and signature generation (key sizes up to 4096 bit).
- Package RSA Key Generation: Generation of RSA key pairs and public key computation (key sizes up to 4096 bit).
- Package ECC over GF(p):
  - The ECDSA (ECC over GF(p)) algorithm can be used for signature generation and signature verification.
  - The ECDSA (ECC over GF(p)) key generation algorithm can be used to generate ECC over GF(p) key pairs for ECDSA.
  - The ECDH (ECC Diffie-Hellman) key exchange algorithm can be used to establish cryptographic keys. It can be also used as secure point multiplication.
  - Provide secure point addition for Elliptic Curves over GF(p).
  - Provide curve parameter verification for Elliptic Curves over GF(p).
  - The TOE supports various key sizes for ECC over GF(p) up to a limit of 640 bits for signature generation, key pair generation and key exchange. For signature verification the TOE supports key sizes up to a limit of 640 bits.
- Package SHA:
  - The SHA-1, SHA-224, SHA-256, SHA-384, and SHA-512 algorithms can be used for different purposes such as compute hash values in the course of digital signature creation and key verification.
- The crypto library implements the KeyStore feature for a secure key management in RAM (see Section 7.2.2.2 for details).
- The crypto library further implements secure move, copy, and compare operations. Even though the TOE does not implement the full PACE protocol, it provides basic support for the implementation of this protocol in the IC Embedded Software via these secure operations.

6. The Flashloader OS is an optional module and stored in a memory segments assigned to SM of Logical Card B and cannot be directly accessed by the Security IC Embedded Software. It is located in ROM and FLASH. One-time executed code is located in FLASH and is removed after use. The freed up memory is then available for the Security IC Embedded Software.
   The Flashloader OS supports the download of code and data to Flash by the Composite Product Manufacturer before Operational Usage (e.g. during development). This functionality can be made unavailable after usage. When the Flashloader OS module is available, the Library Interface, the N7121 Crypto Library and the System Mode OS become mandatory. All logical dependencies of the IC Dedicated Support Software are described in the definitions above.

Note: *Both cryptographic libraries are provided as a library rather than as a monolithic program, and hence a user of the library may include only those functions that are actually required – it is not necessary to include all cryptographic functions of the library in every Security IC Embedded Software. For this purpose, Table 4 defines different packages of the crypto libraries which can be included in the customer application. For example, it is possible to use the package ECC over GF(p) only in case of the N7121 Crypto Library. The inter-dependencies of the different packages are resolved in Table 4.*

With respect to Application note 32 of the [PP], the physical location of the Security IC Embedded Software can be either in ROM or in Flash. The Security IC Embedded Software itself is not in the scope of this evaluation.

All logical dependencies of the IC Dedicated Support Software are described in the definitions above.

1.4.4 Security during Development and Production

The Security IC product life-cycle is scheduled in phases as introduced in [PP]. IC Development as well as IC Manufacturing and Testing, which are Phases 2 and 3 of the life-cycle, are part of the evaluation. Phase 4 the IC Packaging is also part of the evaluation. The Security IC is delivered at the end of Phase 3 or Phase 4 in the life-cycle (Application Note 1 of [PP]). The development and production environment of the TOE ranges from Phase 2 to TOE Delivery.

With respect to Application Note 3 in [PP] the TOE supports the authentic delivery using the FabKey feature. For further details refer to the data sheet [DSheet] and the guidance and operation manual [GOM].

During the design and the layout process only personnel involved in the specific development project for an IC have access to sensitive data. Different teams are responsible for the design data and for customer related data. The production of wafers includes two different steps regarding the production flow. In the first step the wafers are produced with the fixed masks independent of the NCN or CCN. After that step the wafers are completed with the product type specific data, including ROM and Flash Code, and data (if applicable) as identified by NCN and CCN. The test process of every die is performed in CC certified test centers. Delivery processes between the involved sites provide accountability and traceability of the TOE. The TOE is provided in form of sawn wafers, modules, inlays or packages depending on the individual commercial type.

1.4.5 TOE intended usage

The end-consumer environment of the TOE is Phase 7 of the Security IC product life-cycle as defined in [PP]. In this phase the Security IC product is in usage by the end-consumer. Its method of use now depends on the Security IC Embedded Software. The Security ICs including the TOE can be used to perform various functions in a wide range of applications. Examples are identity cards, Banking Cards, Pay-TV, Health cards and Transportation cards. The end-user environment covers a wide spectrum of very different functions, thus making it difficult to monitor and avoid abuse of the TOE. The TOE is intended to be used in an insecure environment, which does not protect against threats.

The device is developed for high-end safeguarded applications, and is designed to be suited for embedding into chip cards with various possible communication interfaces, for example ISO/IEC 7816, contactless applications according to ISO/IEC 14443 or others. Usually a Security IC (e.g. a smartcard) is assigned to a single individual only, but it may also be used by multiple applications in a multi-provider environment. Therefore the TOE might store and process secrets of several systems, which must be protected from each other. The TOE then must meet security requirements for each single security module.

Secret data shall be used as input for calculation of authentication data, calculation of signatures and encryption of data and keys.

In development and production environment of the TOE the Security IC Embedded Software developer and system integrators such as the terminal software developer may use samples of the TOE for their testing purposes. It is not intended that they are able to change the behavior of the Security IC in another way than an end-consumer. The user environment of the TOE ranges from TOE delivery to Phase 7 of the Security IC product life-cycle, and must be a controlled environment up to Phase 6.

Note: *Please note that the phases from TOE Delivery to Phase 7 of the Security IC Product life-cycle are not part of the TOE construction process in the sense of this Security Target. Information about these phases is just included to describe how the TOE* *is used after its construction. Nevertheless such security functionality of the TOE, that is independent of the Security IC Embedded Software, is active at TOE Delivery and cannot be disabled by the Security IC Embedded Software in the following phases.*

1.4.6 Interface of the TOE

The electrical interface of the N7121 are the pads to connect the lines power supply, ground, reset input and clock input. The TOE provides a general purpose I/O interface which allows direct access to the internal SFR bus. This is an external interface that can be used to connect an I2C interface for instance without affecting the certification.

The TOE implements conventional contact-based and contactless interfaces (ISO/IEC 7816 contact interface with UART and ISO/IEC 14443A contactless interface). The availabilty of these interfaces depends on the actual configuration of the TOE.

The logical interface of the TOE depends on the CPU mode and the associated software.

- Upon every start-up the Boot Software is executed in Super System Mode. This software initializes and configures the TOE. This comprises the selection of IC Dedicated Test Software (before TOE delivery) and of Security IC Embedded Software (after TOE delivery). Only in case the configuration option ’Enable Chip Health Mode’ is enabled, starting of built-in self-test routines and read-out of TOE identification items is supported. If this configuration option is disabled, the Boot Software provides no interface. In this case there is no possibility to interact with this software.
- Before TOE delivery the logical interface is defined by the IC Dedicated Test Software. This IC Dedicated Test Software is executed in Super System Mode and comprises the test operating system used for production testing. IC Dedicated Test Software is embedded in the Test Software.
- In System Mode and User Mode (after TOE Delivery) the software interface is the set of instructions, the bits in the special function registers that are related to these modes and the physical address map of the CPU including memories. The access to the special function registers as well as to the memories depends on the TOE mode configured by the Security IC Embedded Software.

Note: *The logical interface of the TOE that is visible on the electrical interface after TOE Delivery is based on the Security IC Embedded Software developed by the software developer. The identification and authentication of the user in System Mode or User Mode must be controlled by the Security IC Embedded Software.*

Note: *The chip surface can be seen as an interface of the TOE, too. This interface must be taken into account regarding environmental stress e.g. like temperature and in the case of an attack, for which the attacker manipulates the chip surface.*

Note: *An external voltage and timing supply as well as a logical interface are necessary for the operation of the TOE. Beyond the physical behavior of the logical interface is defined by the Security IC Embedded Software.*

2 Conformance claims

This Security Target claims to be conformant to the Common Criteria version 3.1:

- *Common Criteria, Part 1: Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and general model*, Version 3.1, Revision 5, April 2017, CCMB-2017-04-001. [CC_Part1]
- *Common Criteria, Part 2: Common Criteria for Information Technology Security Evaluation, Part 2: Security functional components*, Version 3.1, Revision 5, April 2017, CCMB-2017-04-002. [CC_Part2]
- *Common Criteria, Part 3: Common Criteria for Information Technology Security Evaluation, Part 3: Security assurance components*, Version 3.1, Revision 5, April 2017, CCMB-2017-04-003. [CC_Part3]

For the evaluation the following methodology will be used:

- *Common Methodology for Information Technology Security Evaluation, Evaluation Methodology*, Version 3.1, Revision 5, April 2017, CCMB-2017-04-004. [CEM]

This Security Target claims to be **CC Part 2 extended** and **CC Part 3 conformant**. The extended Security Functional Requirements are defined in Section 5.

2.1 Package claim

This Security Target claims conformance to the assurance package EAL6 augmented. The augmentations to EAL6 is ALC_FLR.1. In addition, the Security Target is augmented using the component ASE_TSS.2, which is chosen to include architectural information on the security functionality of the TOE.

The level of evaluation and the functionality of the TOE are chosen in order to allow the confirmation that the TOE is suitable for use within devices compliant with the German Digital Signature Law.

***Note:** The Protection Profile (PP) ”Security IC Platform Protection Profile with Augmentation Packages” [PP] to which this Security Target claims conformance (refer to Section 2.2) requires assurance level EAL4 augmented. The changes, which are needed for EAL6, are described in the relevant sections of this Security Target.*

2.2 PP claim

This Security Target claims strict conformance to the Protection Profile (PP):

- *Security IC Platform Protection Profile with Augmentation Packages*, Version 1.0, registered and certified by Bundesamt für Sicherheit in der Informationstechnik (BSI) under the reference BSI-PP-0084-2014 [PP].

Thus, the concepts are used in the same sense. For the definition of terms refer to [PP]. This chapter does not need any supplement in the Security Target.

This conformance claim includes the following packages of security requirements out of those for Loader defined in the Protection Profile:

- Package “Package 1: Loader dedicated usage in Secured Environment only” Conformant and
- Package “Package2: Loader dedicated for usage by authorized users only” Conformant.

This conformance claim includes the following packages of security requirements out of those for Cryptographic Services defined in the Protection Profile [PP]:

- Package ”TDES” Conformant and
- Package ”AES” Conformant.

If the respective package of the crypto library is available, the additional functionality results in the following change to the conformance claim:

- Package ”TDES” Augmented and
- Package ”AES” Augmented.

Furthermore, if the respective package of the crypto library is available, the additional functionality results in the inclusion of the following conformance claim:

- Package ”Hash functions” Conformant.

The TOE provides additional functionality, which is not covered in [PP]. In accordance with Application Note 4 of [PP] this additional functionality is added using the policy P.Add-Components and P.Add-Crypto-Func (see Section 3.3).

2.3 Conformance claim rationale

According to Section 2.2 this ST claims strict conformance to [PP].

The TOE type defined in Section 1.3.5 is a smartcard controller with IC Dedicated Software. This is consistent with the TOE definition for a Security IC in Section 1.2.2 of [PP]. The sections within this document where Security Problem Definitions, Security Objectives and Security Functional Requirements (SFR) are defined, clearly state which of these items are taken from the Protection Profile and which are added in this Security Target. Moreover, all additionally stated items in this Security Target do not contradict the items included from the PP (see the respective sections in this document). The operations done for the SFRs taken from the PP are also clearly indicated.

The evaluation assurance level claimed for this TOE is shown in Section 6.2 to include respectively exceed the requirements claimed by the PP (EAL4+).

These considerations show that the Security Target correctly claims conformance to [PP].

3 Security problem definition

This chapter lists the assets, threats, assumptions and organizational security policies from [PP] and describes extensions to these elements in detail.

3.1 Description of assets

All assets, which are defined in Section 3.1 of the [PP], are related to standard functionality. They are applied in this Security Target. These assets are:

- integrity and confidentiality of User Data stored and in operation,
- integrity and confidentiality of Security IC Embedded Software, stored and in operation,
- correct operation of the Security Services provided by the TOE for the Security IC Embedded Software, and
- deficiency of random numbers.

To be able to protect these assets the TOE shall protect its Security Functionality. Therefore critical information on the TOE shall be protected. Critical information includes:

- logical design data, physical design data, IC Dedicated Software, as well as
- initialization data and pre-personalization data, Security IC Embedded Software, specific development aids, test and characterization related data, material for software development support, and photo masks.

Note: *Note that the keys for cryptographic calculations using security services of the TOE are treated as User Data.*

3.2 Threats

The Threats defined in Protection Profile are used for this ST without change. Therefore, see [PP] for their definitions. A complete list of Threats defined in [PP] is given in the following table:

Table 5. Threats defined in Protection Profile

Name	Title
T.Leak-Inherent	Inherent Information Leakage
T.Phys-Probing	Physical Probing
T.Malfunction	Malfunction due to Environmental Stress
T.Phys-Manipulation	Physical Manipulation
T.Leak-Forced	Forced Information Leakage
T.Abuse-Func	Abuse of Functionality
T.RND	Deficiency of Random Numbers

In compliance with Application Note 4 in [PP], the TOE provides additional functionality to protect against threats that appear when the TOE is used for multiple applications.

The TOE provides the Security IC Embedded Software running in System Mode with control of access to memories and hardware components by different applications running in User Mode. In this context, User Data of different applications is stored to such memory and processed by such hardware components. The Security IC Embedded Software controls all these User Data. Any access to User Data assigned to one application by another application contradicts separation between different applications and is considered as a threat.

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

T.Unauthorised-Access **Unauthorized Memory or Hardware Access**

Adverse action: An attacker may try to read, modify or execute code or data stored in restricted memory areas. An attacker may try to access or operate hardware resources that are restricted by executing code that accidentally or deliberately accesses these restricted hardware resources. Any code executed or data used in System Mode or User Mode may accidentally or deliberately access code or User Data of other applications. Any code executed or data used in System Mode or User Mode may accidentally or deliberately access hardware resources that are restricted to other applications

Threat agent: Attacker having high attack potential and access to the TOE.

Asset: Code executed by and data belonging to the IC Dedicated Support Software running in Super System Mode or Test Mode as well as code executed by and data belonging to the Security IC Embedded Software.

Application Note: In case the NXP System Mode OS is available, this Threat also covers an attacker who may try to use malicious code placed in the User Mode of Card A or B to modify the correct behavior of the IC Dedicated Software or the Security IC Embedded Software as well as read or modify code or data belonging to the Security IC Dedicated Software or the Security IC Embedded Software.

T.Malicious-UM-Application **Malicious code running in UM of Card A (optional)**

Adverse action: An attacker may try to use malicious code placed in User Mode of logical card A to modify the correct behavior of the Security IC Dedicated Software or the Security IC Embedded Software as well as read or modify code or data belonging to the Security IC Dedicated Software or the Security IC Embedded Software.

Threat agent Attacker having high attack potential and access to the TOE.

Asset: Code executed by and data belonging to the the Security IC Dedicated Software and the Security IC Embedded Software.

Application Note: This threat is only applicable if the logical card A is available.

Restrictions of access to memories and hardware resources, which are available to the Security IC Embedded Software, must be defined and implemented by the security policy of the Security IC Embedded Software based on the specific application context.

3.3 Organizational security policies

The Security Policies defined in [PP] are used for this ST without change. Therefore, see [PP] for their definitions. A complete list of Threats defined in [PP] is given in the following table:

Table 6. Security policies defined in the Protection Profile

Name	Title
P.Process-TOE	Identification during TOE Development and Production
P.Lim_Block_Loader	Limiting and Blocking the Loader Functionality
P.Ctlr_Loader	Controlled usage to Loader Functionality
P.Crypto-Service	Cryptographic services of the TOE

In compliance with Application Note 5 in the [PP], this Security Target defines additional security policies as detailed below.

The TOE provides specific security functionality, which can be used by the Security IC Embedded Software. This specific security functionality is not derived from threats identified for the TOE. Instead, the Security IC Embedded Software decides how to use this security functionality to protect from threats for the composite product. Thus, security policy P.Add-Components is defined as follows.

P.Add-Components **Additional Specific Security Components**

The TOE shall provide the following additional security functionality to the Security IC Embedded Software:

- self-tests, and
- integrity support of data stored to NVM.

P.Add-Crypto-Func **Additional Cryptographic Functionality (optional)**

The TOE shall provide the following additional cryptographic functionality to the Security IC Embedded Software:

- PUF functionality,
- RSA encryption, decryption, signature generation, signature verification, message encoding and signature encoding,
- RSA public key computation,
- RSA key generation,
- ECDSA (ECC over GF(p)) signature generation and verification,

- ECC over GF(p) key generation,
- ECDH (ECC Diffie-Hellmann) key exchange,
- ECC over GF(p) point addition,
- ECC over GF(p) curve parameter verification, and
- SHA-1, SHA-224, SHA-256, SHA-384 and SHA-512 Hash Algorithms.

Note: *This policy depends on the TOE configuration and the availability of the N7121 Crypto Library.*

3.4 Assumptions

All assumptions defined in Section 3.4 of [PP] are valid for this Security Target:

Table 7. Assumptions defined in the Protection Profile

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

In compliance with Application Notes 6 and 7 of [PP], this Security Target defines two additional assumptions as follows.

A.Check-Init

Check of initialization data by the Security IC Embedded

The Security IC Embedded Software must provide a function to check initialization data. Such data is defined by the Composite Product Manufacturer and injected by the TOE Manufacturer into the non-volatile memory to provide the ability to identify and trace the TOE.

A.Key-Function

Usage of Key-dependent Functions

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

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

4 Security objectives

4.1 Security objectives of the TOE

All Security Objectives of the TOE, which are defined in [PP] are applied to this Security Target. This also comprises the Security Objectives defined in the functional packages which are claimed in Section 2.2. The following table lists these Security Objectives of the TOE:

Table 8. Security objectives ot the TOE defined in the Protection Profile

Name	Title
O.Leak-Inherent	Protection against Inherent Information Leakage
O.Phys-Probing	Protection against Physical Probing
O.Malfunction	Protection against Malfunctions
O.Phys-Manipulation	Protection against Physical Manipulation
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 (optional)	Access control and authenticity for the Loader
O.TDES	Cryptographic service Triple-DES
O.AES	Cryptographic service AES
O.SHA (optional)	Cryptographic service Hash functions

The objective O.Ctrl_Auth_Loader depends on the current state of the Flash Loader. In case the Flash Loader is blocked (after usage or as the Flash Loader is deactivated following Table 3), the respective functionality is not available anymore. The objectives O.AES and O.TDES depend on the availability of the AES and DES coprocessor which can be deactivated following Table 3.

In compliance with Application Notes 8 and 9 of [PP], additional Security Objectives for the TOE are defined below based on additional functionality provided by the TOE.

O.NVM-Integrity

Integrity Support of data stored to NVM

The TOE shall provide detection and correction of failures in NVM memories to support integrity of contents stored there.

O.Access-Control

Access Control to Memories and Special Function Registers

The TOE shall control access of CPU instructions to memory areas depending on memory partitioning by the MMU and based on the CPU modes Super System Mode, System Mode and User Mode. In Super System Mode, System Mode and User Mode, the TOE shall control access also based on configuration. In User Mode, the TOE shall control access also based on memory segments, which are configured in System Mode when implementing a memory management scheme. This control shall be individual to each memory segment and consider different access rights.

The only way to change the TOE mode shall be restricted as well. The change to a higher or less privileged mode shall be done by using explicit instructions or by interrupts triggered by hardware peripherals of the TOE. The TOE shall further provide mechanisms to return to the previous mode.

Furthermore, the TOE shall control access of CPU instructions to RAM based segment descriptors depending on the purpose of the descriptor and based on TOE modes. The TOE shall provide the System Mode with the ability to configure access rights for the User Mode to these descriptors.

O.Self-Test

Self-Test

The TOE shall include functionality to perform a self-test to detect physical manipulation.

O.PUF (optional)

Sealing/Unsealing user data

The TOE shall provide PUF functionality that supports sealing/unsealing of User Data. Using this functionality, the User Data can be sealed within the TOE and can only be unsealed by the same TOE that the User Data was sealed on. The PUF functionality comprises import/export of data, encryption/decryption of data and calculation of a MAC as a PUF authentication value.

Note: *The PUF functionality provided by the TOE shall only be active if explicitly configured by the Security IC Embedded Software.*

Note: *This objective requires the availability of the PUF which can be deactivated depending on the ordered TOE configuration. Furthermore, encryption and decryption requires the availability of the AES coprocessor which can be deactivated via Post-Delivery Configuration.*

O.Secure-User-Mode-Box (optional)

Secure User Mode Box Firewall

The TOE shall provide specific separation between the Secure UM Box code and other parts of the TOE. The separation shall comprise software execution and data access.

Note: *This objective is only applicable if the customer ordered an NXP application running in User Mode of* *logical card A and relies on the NXP System Mode OS running in System Mode of logical card A.*

O.RSA

RSA Functionality (optional)

The TOE includes functionality to provide encryption, decryption, signature creation, signature verification, message encoding and signature encoding using the RSA algorithm. Furthermore, the TOE provides functionality to compute a public RSA key from a given private RSA key as well as RSA key-pair generation.

Note: *This objective requires the availability of the N7121 Crypto Library.*

O.ECC

Elliptic-Curve Cryptography over GF(p) (optional)

The TOE provides signature generation and verification, Diffie-Hellmann key exchange, each using the ECC over GF(p) algorithm. It further includes functionality to generate ECC over GF(p) key pairs.

Note: *This objective requires the availability of the N7121 Crypto Library.*

4.2 Security objectives of the security IC embedded software development

All security objectives for the Security IC Embedded Software development Environment, which are defined in [PP], are applied to this Security Target.

Table 9. Security Objectives of the Security IC Embedded Software Development defined in Protection Profile

Name	Title
OE.Resp-Appl	Treatment of User Data

Clarification related to OE.Resp-Appl:

By definition cipher or plain text data and cryptographic keys are User Data. The Security IC Embedded Software shall treat these data appropriately, use only proper secret keys (chosen from a large key space) as input for the cryptographic function of the TOE and use keys and functions appropriately in order to ensure the strength of cryptographic operation. This means that keys are treated as confidential as soon as they are generated. The keys must be unique with a very high probability, as well as cryptographically strong. If keys are imported into the TOE and/or derived from other keys, quality and confidentiality must be maintained. This implies that appropriate key management has to be realized in the environment.

In case the Security IC Embedded Software operates multiple applications on the TOE, OE.Resp-Appl must also be met. The Security IC Embedded Software must not disclose security relevant User Data of one application to another application when processed in or stored to the TOE.

4.3 Security objectives for the Operational Environment

All Security Objectives for the Operational Environment of the TOE, which are defined in [PP] are applied to this Security Target. This also comprises the Security Objectives for the Operational Environment defined in the functional packages which are claimed in Section 2.2. The following table lists these Security Objectives of the TOE:

Table 10. Security objectives for the Operational Environment

Name	Title
OE.Process-Sec-IC	Protection during composite product manufacturing
OE.Lim_Block_Loader	Limitation of capability and blocking the Loader
OE.Loader_Usage (optional)	Secure communication and usage of the Loader

The following additional security objectives for the operational environment are defined in this Security Target.

The following security objective for the operational environment derives from assumption A.Check-Init. The TOE provides specific functionality that requires the TOE Manufacturer to implement measures for unique identification of the TOE. Security objective OE.Check-Init is defined to allow for such a TOE specific implementation.

OE.Check-Init

Check of initialization data by the Security IC Embedded Software

To ensure the receipt of the correct TOE, the Security IC Embedded Software shall check a sufficient part of the pre-personalization data. This shall include at least the FabKey data that is agreed between the customer and the TOE Manufacturer.

4.4 Security objectives rationale

Section 4.4 of [PP] provides a rationale how the threats, organisational security policies and assumptions are addressed by the Security Objectives defined in [PP]. The following table summarizes how Threats, Organizational Security Policies and Assumptions defined in this ST in extension to [PP] are addressed by Security Objectives defined in the PP and ST, respectively.

Table 11. Security Objectives (PP and ST) vs. Security Problem Definition (PP and ST)

Security Problem Definition	Security Objective	Note
T.Leak-Inherent	O.Leak-Inherent	--
T.Phys-Probing	O.Phys-Probing	--
T.Malfunction	O.Malfunction **O.Self-Test**	--
T.Phys-Manipulation	O.Phys-Manipulation **O.Self-Test**	--
T.Leak-Forced	O.Leak-Forced	--
T.Abuse-Func	O.Abuse-Func	--

Security Problem Definition	Security Objective	Note
T.RND	O.RND	--
P.Process-TOE	O.Identification	Phases 2–3
A.Process-Sec-IC	OE.Process-Sec-IC	Phases 4–6
A.Resp-Appl	OE.Resp-Appl	Phase 1
P.Lim_Block_Loader	O.Cap_Avail_Loader OE.Lim_Block_Loader	--
P.Ctlr_Loader	O.Ctrl_Auth_Loader OE.Loader_Usage	--
P.Crypto-Service	O.TDES O.AES O.SHA	--
T.Unauthorised-Access	**O.Access-Control**	--
T.Malicious-UM-Application (optional)	**O.Secure-User-Mode-Box (optional)**	--
P.Add-Components	**O.Self-Test** **O.NVM_Integrity**	--
P.Add-Crypto-Func (optional)	**O.PUF (optional)** **O.RSA (optional)** **O.ECC (optional)**	--
A.Check-Init	**OE.Check-Init**	--
A.Key-Function	OE.Resp-Appl	--

In the table above, bold text is used to indicate threats, OSPs, assumptions, and objectives which are added to this ST in extension to the PP.

The following table provides rationales for the assignments of Security Objectives to Threats, and Policies which are not already provided in [PP].

Table 12. Rationales for the assignments between the Security problem definition and the Security objectives not already covered in the Protection Profile

Security problem definition	Security objective	Rationale
T.Malfunction	**O.Self-Test**	This objective requires that the TOE provides self-testing features for security critical components, thus contributing to cover this threat.
T.Phys-Manipulation	**O.Self-Test**	This objectives requires that the TOE provides self-testing features for security critical components, thus contributing to cover this threat.

Security problem definition	Security objective	Rationale
T.Unauthorised-Access	**O.Access-Control**	The TOE has to enforce memory partitioning with address mapping and control of access to memories in System Mode and User Mode. Access rights in User Mode must be explicitly granted by Security IC Embedded Software running in System Mode. Thus, security violations caused by accidental or deliberate access to restricted data, code and shared hardware resources can be prevented. The TOE further has to enforce control of access to Special Function Registers in System Mode and User Mode. Access rights in User Mode must be explicitly granted by code running in System Mode. Thus, security violations caused by accidental or deliberate access to restricted data, code and shared hardware resources can be prevented.
T.Malicious-UM-Application (optional)	**O.Secure-User-Mode-Box (optional)**	This objectives enforces the Secure User-Mode Box, thus contributing to cover this threat.
P.Add-Components	**O.Self-Test**	This objective covers the security policy because it requires the TOE to partly implement the functionality as required by the security policy.
	O.NVM-Integrity	This objective covers the security policy because it requires the TOE to partly implement the functionality as required by the security policy.
P.Add-Crypto-Func (optional)	**O.PUF (optional)**	This objective covers the security policy because it requires the TOE to partly implement the functionality as required by the security policy.
	O.RSA (optional)	This objective covers the security policy because it requires the TOE to partly implement the functionality as required by the security policy.
	O.ECC (optional)	This objective covers the security policy because it requires the TOE to partly implement the functionality as required by the security policy.
A.Check-Init	**OE.Check-Init**	This objective requires the Security IC Embedded Software developer to implement a function as stated in this assumption.

Security problem definition	Security objective	Rationale
A.Key-Function	OE.Resp-Appl	The definition of this objective of the [PP] is further clarified in this Security Target: By definition cipher or plain text data and cryptographic keys are User Data. So, the Security IC Embedded Software will protect such data if required and use keys and functions appropriately in order to ensure the strength of cryptographic operation. Quality and confidentiality must be maintained for keys that are imported and/or derived from other keys. This implies that appropriate key management has to be implemented in the environment. In addition, the treatment of User Data comprises the implementation of a multi-application operating system that does not disclose security relevant User Data of one application to another one. These measures make sure that the assumption A.Key-Function is still covered by this objective.

The justification of the additional policy and the additional assumptions show that they do not contradict to the rationale already given in the Protection Profile for the assumptions, policy and threats defined there.

5 Extended components definition

The underlying Protection Profile [PP] contains extended components. This Security Target does not define further extended components.

6 Security requirements

This chapter defines the security requirements that shall be met by the TOE. These security requirements are composed of the Security Functional Requirements (SFR) and the Security Assurance Requirements (SAR) that the TOE must meet in order to achieve its security objectives. CC allows several operations to be performed on security requirements (on the component level); refinement, selection, assignment, and iteration are defined in Section 8.1 of [CC_Part1]. These operations are used in the [PP] and in this Security Target, respectively.

- The refinement operation is used to add details to requirements, and thus, further intensifies a requirement. Refinements are indicated as **bold text**.
- The selection operation is used to select one or more options provided by the PP or CC in stating a requirement. Selections having been made are denoted as *italic text*.
- The assignment operation is used to assign a specific value to an unspecified parameter, such as the length of a password. Assignments having been made are denoted as underlined text.
- The ST applies further refinements by deleting specific words. These refinements are indicated by $_{\text{subscript text}}$. These refinements do not affect the meaning of the SFRs and are only applied for grammatical reasons.
- The iteration operation is used when a component is repeated with varying operations. It is denoted by the same notation used in [PP], i.e., a slash followed by the iteration indicator. Whenever an element in [PP] contains an operation that is left uncompleted, the Security Target has to complete that operation.

***Note:** Please note that this ST does not indicate the operations already performed in [PP]. Therefore, this ST only highlights those operations which are left open in the [PP]. If an SFR was not taken from the certified PP but from CC Part 2, the ST identifies all operations required by [CC_Part2].*

Furthermore, the following sections provide application notes for each SFR as an informative text. These notes are used to indicate the dependency of each SFR to the configuration of the TOE.

6.1 Security functional requirements

All Security Functional Requirements (SFRs) of the TOE are presented in the following sections to support a better understanding of the combination of the PP and this Security Target. With respect to Application Note 12 in [PP], it is clearly stated, which subset of SFRs is taken from the underlying protection profile or its functional packages and which are newly introduced.

6.1.1 Security Functional Requirements of the PP

FRU_FLT.2 **Limited fault tolerance**

FRU_FLT.2.1 The TSF shall ensure the operation of all the TOE's capabilities when the following failures occur: exposure to operating conditions which are not detected according to the requirement Failure with preservation of secure state (FPT_FLS.1).

Application Note:

The term ”failure” above means ”circumstances”. The TOE prevents failures for the ”circumstances” defined above (refinement from [PP]). With respect to Application Notes 15 in [PP], generation of additional audit data is not defined.

This SFR is in place for each TOE configuration.

FPT_FLS.1 **Failure with preservation of secure state**

FPT_FLS.1.1

The TSF shall preserve a secure state when the following types of failures occur: exposure to operating conditions which may not be tolerated according to the requirement Limited fault tolerance (FRU_FLT.2) and where therefore a malfunction could occur.

Application Note:

The term “failure” above also covers “circumstances”. The TOE prevents failures for the “circumstances” defined above (refinement from [PP]). With respect to Application Note 15 in [PP], generation of additional audit data is not defined.

This SFR is in place for each TOE configuration.

FMT_LIM.1 **Limited capabilities**

FMT_LIM.1.1

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

Application Note:

This SFR is in place for each TOE configuration.

FMT_LIM.2 **Limited availability**

FMT_LIM.2.1

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

Application Note:

This SFR is in place for each TOE configuration.

FAU_SAS.1 **Audit storage**

FAU_SAS.1.1 The TSF shall provide the test process before TOE Delivery with the capability to store the *Initialisation Data*, *Pre-personalisation Data* *and customer-specific Data* in the Flash.

Application Note: With respect to the Application Notes 16 and 17 of [PP] , the TOE provides the necessary data for identification and the SFR states the storage location.

This SFR is in place for each TOE configuration.

FDP_SDC.1 **Stored data confidentiality**

FDP_SDC.1.1 The TSF shall ensure the confidentiality of the information of the user data while it is stored in the ROM, RAM and Non-Volatile Memory.

Application Note: This SFR is in place for each TOE configuration.

FDP_SDI.2 **Stored data integrity monitoring and action**

FDP_SDI.2.1 The TSF shall monitor user data stored in containers controlled by the TSF for modification, deletion, repetition or loss of data on all objects, based on the following attributes: integrity check information associated with the data stored in memories.

FDP_SDI.2.2 Upon detection of a data integrity error, the TSF shall perform an error correction if possible and a Security Reset if not.

Application Note: With respect to the Application Notes 18 of the [PP], the necessary operations were performed.

This SFR is in place for each TOE configuration.

FPT_PHP.3 **Resistance to physical attack**

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

Application Note: The TSF will implement appropriate mechanisms to continuously counter physical manipulation and physical probing. Due to the nature of these attacks (especially manipulation) the TSF cannot detect attacks on all of its elements. Therefore, permanent protection against these attacks is required ensuring that security functional requirements are enforced. Hence, “automatic response” means here (i) assuming that there might be an attack at any time and (ii) countermeasures are provided at any time (refinement from [PP]).

This SFR is in place for each TOE configuration.

FDP_ITT.1 **Basic internal transfer protection**

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

Application Note: The different memories, the CPU and other functional units of the TOE (e.g. a cryptographic co-processor) are seen as physically-separated parts of the TOE.

This SFR is in place for each TOE configuration.

FPT_ITT.1 **Basic internal TSF data transfer protection**

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

Application Note: The different memories, the CPU and other functional units of the TOE (e.g. a cryptographic co-processor) are seen as separated parts of the TOE (see [PP]).

This SFR is in place for each TOE configuration.

FDP_IFC.1 **Subset information flow control**

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

Application Note: The different memories, the CPU and other functional units of the TOE (e.g. a cryptographic co-processor) are seen as physically-separated parts of the TOE (see [PP]).

This SFR is in place for each TOE configuration.

FCS_RNG.1/PTG.2 **Random number generation – PTG.2**

FCS_RNG.1.1/PTG.2 The TSF shall provide a physical random number generator that implements:

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

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

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

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

(PTG.2.5) - The online test procedure checks the quality of the raw random number sequence. It is triggered *at regular intervals or continuously*. The online test is suitable for detecting non-tolerable statistical defects of the statistical properties of the raw random numbers within an acceptable period of time.

FCS_RNG.1.2/PTG.2

The TSF shall provide *octets of bits* that meet:

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

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

Application Note:

Application Note 21 in the [PP] refers to for examples for the security capabilities and quality metrics used in some national certification schemes.

The definition of the SFR FCS_RNG.1/PTG.2 was taken from [KS2011], which is identical to the definition found in the [PP], as the TOE is certified in the German Common Criteria scheme.

In accordance with Application Note 44 of the [PP], the assignment for additional standard statistical test suite in clause (PTG.2.6) may be empty.

The Shannon entropy 0.997 per internal random bit compares to 7.976 per octet.

The Shannon entropy is computed as $E = -\sum_{i=0}^{255} p_i \log_2 p_i$, where p_i is the probability that the byte $(b_7, b_6, \dots, b_0)$ is equal to i as binary number. The value 7.976 is assigned due to the requirements of [AIS31].

The results of the internal test sequence are provided to the Security IC Embedded Software as a pass or fail criterion.

Application Note:

The N7121 Crypto Library supports statistical test as required in this SFR. The IC Embedded Software has to take care of testing the random numbers generated by the RNG by means of software statistical test. Therefore, this SFR is only partially fulfilled by the hardware TOE. If the Crypto Library is used, the required test functionality is available. In this case, the SFR is completely fulfilled.

6.1.2 Flash Loader (partly optional)

The following table describes the subjects and objects of the Loader policy.

Table 13. Subjects, objects as well as related operations and attributes of the Loader Policy

Identifier	Description
Subjects	
Download User	User Role to download data, verify data and erase data in memory areas.
Key Change User	User Role to update and verify keys.
Developer Mode User	User Role to switch the life cycle to Pre-Release.
Production Mode User	User Role to switch the life cycle to Release.
Card Operating System	The Card Operating System.
Unauthorized User	An unauthorized user.
Objects as well as related operations and attributes	
Life-Cycle State	Life Cycle State of the Loader. The only available operation is: • Switch – Switch from Download to Pre-Release, from Pre-Release to Download or from Download to Release. The available attributes of Life Cycle State are: • Download – Initial Life-Cycle State of the TOE which allows download operations. • Pre-Release – Previously downloaded code can be executed. Furthermore, it is possible to return to Life-Cycle State Download. • Release – Final state of the Flash Loader after permanent blocking. No download operations can be performed anymore. It is not possible to switch back to another state of the Life-Cycle State.
Keys	Cryptographic keys used to identify users. The only available operation is: • Update – Update or verify a key. The only attribute is: • Permissions – Permissions associated with one key to identify subjects.

Identifier	Description
Memory Segments	Memory segments to which data or code can be downloaded. The available operations are: • Download – Download data to a memory segment. • Verify – Verifies the data downloaded to a memory segment. • Erase – Erase data within a memory segment. No attributes available.
User Data	User Data to be stored to, verified in, or removed from Memory Segments. Operations are already covered by the object Memory Segments.

6.1.2.1 Loader Package 1 defined in the PP

FMT_LIM.1/Loader **Limited capabilities – Loader**

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/Loader)” the following policy is enforced

Deploying Loader functionality after switching to Life Cycle State Release does not allow stored User Data to be disclosed or manipulated by Unauthorized User.

Application Note: In Life Cycle State Release, no download operations can be performed anymore. This corresponds to blocking the Flash Loader permanently.

This SFR is in place for each TOE configuration. In case the Flash Loader is not selected in the TOE configuration, its functionality is blocked following this SFR.

FMT_LIM.2/Loader **Limited availability – Loader**

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/Loader)” the following policy is enforced

The TSF prevents deploying the Loader functionality after switching to Life Cycle State Release.

Application Note: In Life Cycle State Release, no download operations can be performed anymore. This corresponds to blocking the Flash Loader permanently.

This SFR is in place for each TOE configuration. In case the Flash Loader is not selected in the TOE configuration, its functionality is blocked following this SFR.

6.1.2.2 Loader Package 2 defined in the PP (optional)

The following SFR depend on the configuration of the TOE. In case the Flash Loader is set to be available, the following SFRs describe the use of the Flash Loader. As soon as the Flash Loader is blocked (which corresponds to the situation when the TOE is configured without Flash Loader), the SFRs of Loader Package 1 address the blocking of the loader functionality.

FTP_ITC.1/Loader

Inter-TSF trusted channel (optional)

FTP_ITC.1.1/Loader

The TSF shall provide a communication channel between itself and Download User, Key Change User, Developer Mode User, and Production Mode User that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification and disclosure.

FTP_ITC.1.2/Loader

The TSF shall permit another trusted IT product to initiate communication via the trusted channel.

FTP_ITC.1.3/Loader

The TSF shall initiate communication via the trusted channel for deploying Loader functionality as described in FDP_ACF.1/Loader.

Application Note:

In addition to the required operations, this ST also performs an iteration on this SFR for consistency reasons.

This SFR is in place as long as the Flash Loader is not blocked permanently by setting its Life Cycle State to Release.

FDP_UCT.1/Loader

Basic data exchange confidentiality (optional)

FDP_UCT.1.1/Loader

The TSF shall enforce the Loader SFP to receive User Data in a manner protected from unauthorised disclosure.

Application Note:

In addition to the required operations, this ST also performs an iteration on this SFR for consistency reasons.

This SFR is in place as long as the Flash Loader is not blocked permanently by setting its Life Cycle State to Release.

FDP_UIT.1/Loader

Data exchange integrity (optional)

FDP_UIT.1.1/Loader

The TSF shall enforce the Loader SFP to receive User Data in a manner protected from modification, deletion, insertion errors.

FDP_UIT.1.2/Loader

The TSF shall be able to determine on receipt of User Data, whether modification, deletion, insertion has occurred.

Application Note:

In addition to the required operations, this ST also performs an iteration on this SFR for consistency reasons.

This SFR is in place as long as the Flash Loader is not blocked permanently by setting its Life Cycle State to Release.

FDP_ACC.1/Loader

Subset access control – Loader (optional)

FDP_ACC.1.1/Loader

The TSF shall enforce the *Loader SFP* on:

1. the subjects Download User, Key Change User, Developer Mode User, Production Mode User, and Card Operating System,
2. the objects User Data in memory areas which contain Life Cycle State, Keys and Memory Segments,
3. the operation deployment of Loader.

Application Note:

The TOE enforces the Loader SFP by FTP_ITC.1/Loader, FDP_UCT.1/Loader, FDP_UIT.1/Loader, and FDP_ACF.1/Loader to describe additional access control rules.

This SFR is in place as long as the Flash Loader is not blocked permanently by setting its Life Cycle State to Release.

FDP_ACF.1/Loader

Security attribute based access control – Loader (optional)

FDP_ACF.1.1/Loader

The TSF shall enforce the Loader SFP to objects based on the following:

1. the subjects Download User, Key Change User, Developer Mode User, Production Mode User, and Card Operating System with security attributes: none
2. the objects User Data in memory areas which contain Life Cycle State, Keys and Memory Segments with security attributes as listed in Table 13.

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:

1. The Developer Mode User is allowed to switch the Life Cycle State from Download to Pre-Release.
2. The Production Mode User is allowed to switch the Life Cycle State from Download to Release.
3. The Card Operating System is allowed to switch the Life Cycle State from Pre-Release to Download.

FDP_ACF.1.3/Loader

The TSF shall explicitly authorise access of subjects to objects based on the following additional rules:

1. The Download User is allowed to Download, Erase, and Verify Memory Segments if Life Cycle State Download grants this right.
2. The Key Change User is allowed to update Permissions of Keys if Life Cycle State Download grants this right.

FDP_ACF.1.4/Loader

The TSF shall explicitly deny access of subjects to objects based on the following additional rules: as stated in SFR FMT_LIM.2/Loader.

Application Note:

With respect tp Application Note 39 of the [PP], This SFR is in place as long as the Flash Loader is not blocked permanently by setting its Life Cycle State to Release.

6.1.3 Hardware Support for TDES and AES

6.1.3.1 Package TDES defined in PP

The following SFRs address the functionality provided by the TDES coprocessor of the TOE hardware to compute TDES encryption and decryption. With respect to the Functional Package "TDES" of the PP, the functionality provided by the TOE hardware is package conformant.

FCS_COP.1/TDES

Cryptographic operation – TDES

FCS_COP.1.1/TDES

The TSF shall perform *encryption and decryption* in accordance with a specified cryptographic algorithm *TDES in ECB mode* and cryptographic key sizes *112 bit, 168 bit* that meet the following [NIST SP 800-67], [NIST SP 800-38A].

FCS_CKM.4/TDES

Cryptographic key destruction – TDES

FCS_CKM.4.1/TDES

The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method flushing of key registers that meets the following: none.

6.1.3.2 Package "AES" defined in PP

The following SFRs address the functionality provided by the AES coprocessor of the TOE hardware to compute AES encryption and decryption. With respect to the Functional Package "AES" of the PP, the functionality provided by the TOE hardware is package conformant.

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

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

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

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

6.1.4 Further Security Functional Requirements – Cryptographic Library (optional)

6.1.4.1 Library Support for TDES and AES (optional)

The following SFRs address the functionality provided by the N7121 Crypto Library to compute TDES and AES encryption and decryption. With respect to the Functional Packages "TDES" and "AES" of the PP, the functionality provided by the hardware is package augmented.

Therefore, the following set of SFRs is only available if the N7121 Crypto Library is available. It extends the functionality already provided by the hardware as covered in Section 6.1.3.

FCS_COP.1/TDES_LIB **Cryptographic operation – TDES – Crypto Library (optional)**

FCS_COP.1.1/TDES_LIB The TSF shall perform encryption, decryption, MAC generation and MAC verification in accordance with a specified cryptographic algorithm TDES in ECB mode, CBC mode, CBC-MAC mode, Retail-MAC mode, and CMAC mode and cryptographic key sizes 112 bit, 168bit that meets the following:

- [NIST SP 800-67] (TDES),
- [NIST SP 800-38A] (ECB and CBC mode),
- [ISO/IEC 9797-1], Algorithm 1 (CBC-MAC mode),
- [ISO/IEC 9797-1], Algorithm 3 (Retail-MAC mode), and
- [NIST SP 800-38B] (CMAC mode).

Application Note:

This SFR depends on the availability of the N7121 Crypto Library.

FCS_CKM.4/TDES_LIB

Cryptographic Key Destruction – Crypto Library (optional)

FCS_CKM.4.1/TDES_LIB

The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method flushing of key registers that meets the following: none.

Application Note:

The N7121 Crypto Library provides the smartcard embedded software with library calls to perform various cryptographic algorithms that involve keys. Through the parameters of the library calls, the smartcard embedded software provides keys for the cryptographic algorithms. To perform its cryptographic algorithms, the library copies these keys, or a transformation thereof, to the working-buffer (supplied by the smartcard embedded software) and/or the memory/special function registers of the Crypto Library. Depending upon the algorithm the library either overwrites these keys before returning control to the smartcard embedded software or provides a library call to through which the smartcard embedded software can clear these keys. In the case of a separate library call to clear keys the guidance instructs the smartcard embedded software when/how this call should be used.

Clearing of keys that are provided by the smartcard embedded software to the Crypto Library is the responsibility of the smartcard embedded software.

This SFR depends on the availability of the N7121 Crypto Library.

FCS_COP.1/AES_LIB

Cryptographic operation – AES – Crypto Library (optional)

FCS_COP.1.1/AES_LIB

The TSF shall perform encryption, decryption, MAC generation and MAC verification in accordance with a specified cryptographic algorithm AES in ECB mode, CBC mode, CTR mode, CBC-MAC mode, and CMAC mode and cryptographic key sizes 128 bit, 192 bit, and 256 that meets the following:

- [FIPS 197] (AES),
- [NIST SP 800-38A] (ECB, CBC and CTR mode),
- [ISO/IEC 9797-1], Algorithm 1 (CBC-MAC mode), and
- [NIST SP 800-38B] (CMAC mode).

Application Note:

This SFR depends on the availability of the N7121 Crypto Library.

FCS_CKM.4/AES_LIB **Cryptographic Key Destruction – Crypto Library (optional)**

FCS_CKM.4.1/AES_LIB The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method flushing of key registers that meets the following: none.

Application Note: The N7121 Crypto Library provides the smartcard embedded software with library calls to perform various cryptographic algorithms that involve keys. Through the parameters of the library calls, the smartcard embedded software provides keys for the cryptographic algorithms. To perform its cryptographic algorithms, the library copies these keys, or a transformation thereof, to the working-buffer (supplied by the smartcard embedded software) and/or the memory/special function registers of the Crypto Library. Depending upon the algorithm the library either overwrites these keys before returning control to the smartcard embedded software or provides a library call to through which the smartcard embedded software can clear these keys. In the case of a separate library call to clear keys the guidance instructs the smartcard embedded software when/how this call should be used.

Clearing of keys that are provided by the smartcard embedded software to the Crypto Library is the responsibility of the smartcard embedded software.

This SFR depends on the availability of the N7121 Crypto Library.

6.1.4.2 Library Support for Random-Number Generation (optional)

The N7121 Crypto Library provides additional random-number generators as addressed by the following SFRs.

FCS_RNG.1/DRG.4 **Random Number Generation – Hybrid Deterministic (optional)**

FCS_RNG.1.1/DRG.4 The TSF shall provide a hybrid deterministic random number generator that implements:

(DRG.4.1) The internal state of the RNG shall use *PTRNG of class PTG.2 as random source*.

(DRG.4.2) The RNG provides forward secrecy.

(DRG.4.3) The RNG provides backward secrecy even if the current internal state is known.

(DRG.4.4) The RNG provides enhanced forward secrecy *on demand*.

(DRG.4.5) The internal state of the RNG is seeded by an *PTRNG of class PTG.2*.

FCS_RNG.1.2/DRG.4

The TSF shall provide random numbers that meet:

(DRG.4.6) The RNG generates output for which for AES-mode 2^{48} and for TDEA-mode 2^{35} strings of bit length 128 are mutually different with probability at least 1 - 2^{-24}.

(DRG.4.7) Statistical test suites cannot practically distinguish the random numbers from output sequences of an ideal RNG. The random numbers must pass test procedure A.

Application Note:

The definition of this SFR is based on the already performed operations performed in [KS2011]. Therefore, the operations indicated for FCS_RNG.1/DRG.4 are done here not with reference to [PP] where FCS_RNG.1 is defined but with respect to [KS2011].

Similar as in case of FCS_RNG.1/PTG.2, the additional standard statistical test suite in clause DRG.4.7 is left empty.

This SFR depends on the availability of the N7121 Crypto Library.

FCS_RNG.1/PTG.3

Random Number Generation (Hybrid-Physical) (optional)

FCS_RNG.1.1/PTG.3

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 has been detected no random numbers will be output.

(PTG3.2) 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*

(PTG3.3) The online test shall detect non-tolerable statistical defects of the raw random number sequence (i) immediately when the RNG is 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.

(PTG3.4) The online test procedure shall be effective to detect non-tolerable weaknesses of the random numbers soon.

(PTG3.5) The online test procedure checks the raw random number sequence. It is triggered *continuously* . The online test is suitable for detecting non-tolerable statistical defects of the statistical properties of the raw random numbers within an acceptable period of time.

(PTG3.6) 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/PTG.3

The TSF shall provide random numbers that meet:

(PTG3.7) Statistical test suites cannot practically distinguish the random numbers from output sequences of an ideal RNG. The random numbers must pass test procedure A (as defined in [KS2011])

(PTG3.8) The internal random numbers shall *use PTRNG of class PTG.2 as random source for the post-processing*

Application Note:

The definition of this SFR is based on the already performed operations performed in [KS2011]. Therefore, the operations indicated for FCS_RNG.1/DRG.4 are done here not with reference to [PP] where FCS_RNG.1 is defined but with respect to [KS2011].

This SFR depends on the availability of the N7121 Crypto Library.

6.1.4.3 Library Support for RSA (optional)

FCS_COP.1/RSA

Cryptographic operation – RSA (optional)

FCS_COP.1.1/RSA

The TSF shall perform encryption, decryption, signature generation and verification in accordance with a specified cryptographic algorithm RSAEP, RSADP, RSASP1 and RSAVP1 and cryptographic key sizes 512 bits to 4096 bits that meets the following: [PKCS #1].

Application Note:

The security functionality is resistant against side channel analysis and other attacks described in [JIL-ATT-SC]. To fend off attackers with high attack potential an adequate key length must be used (references can be found in national and international documents and standards).

This SFR depends on the availability of the N7121 Crypto Library.

FCS_COP.1/RSA_PAD

Cryptographic operation – RSA message encoding (optional)

FCS_COP.1.1/RSA_PAD

The TSF shall perform message and signature encoding methods in accordance with a specified cryptographic algorithm EME-OAEP and EMSA-PSS and cryptographic key sizes 512 bits to 4096 bits that meets the following: [PKCS #1]: EME-OAEP and EMSA-PSS.

Application Note:

The security functionality is resistant against side channel analysis and other attacks described in [JIL-ATT-SC]. To fend off attackers with high attack potential an adequate key length must be used (references can be found in national and international documents and standards).

This SFR depends on the availability of the N7121 Crypto Library.

FCS_COP.1/RSA_PubExp

Cryptographic operation – RSA public key computation (optional)

FCS_COP.1.1/RSA_PubExp

The TSF shall perform public key computation in accordance with a specified cryptographic algorithm RSAEP, RSADP, RSASP1 and RSAVP1 and cryptographic key sizes 512 bits to 4096 bits that meets the following: [PKCS #1].

Application Note:

The security functionality is resistant against side channel analysis and other attacks described in [JIL-ATT-SC]. To fend off attackers with high attack potential an adequate key length must be used (references can be found in national and international documents and standards).

The computation will result in the generation of a public RSA key from the private key (in CRT format). As this key is implied by the private key, this is not true key generation, and, to prevent duplication in this ST, this has not been included as a separate FCS_CKM.1 SFR.

This SFR depends on the availability of the N7121 Crypto Library.

FCS_CKM.1/RSA

Cryptographic Key Generation – RSA (optional)

FCS_CKM.1.1/RSA

The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm RSA and specified cryptographic key sizes 512 bits to 4096 bits that meet the following: [FIPS 186-4].

Application Note:

The security functionality is resistant against side channel analysis and other attacks described in [JIL-ATT-SC]. To fend off attackers with high attack potential an adequate key length must be used (references can be found in national and international documents and standards).

For the modulus n (n = p*q) the prime numbers p and q generated by the key generator are congruent to 3 modulo 4.

This SFR depends on the availability of the N7121 Crypto Library.

FCS_CKM.4/RSA

Cryptographic Key Destruction – RSA (optional)

FCS_CKM.4.1/RSA

The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method overwrite that meets the following: [ISO 11568-4].

Application Note:

The crypto library provides the smartcard embedded software with library calls to perform various cryptographic algorithms that involve keys. Through the parameters of the library calls, the smartcard embedded software provides keys for the cryptographic algorithms. To perform its cryptographic algorithms, the library copies these keys, or a transformation thereof, to the working-buffer (supplied by the smartcard embedded software) and/or the memory/special function registers of the Crypto Library. Depending upon the algorithm the library either overwrites these keys before returning control to the smartcard embedded software or provides a library call to through which the smartcard embedded software can clear these keys. In the case of a separate library call to clear keys the guidance instructs the smartcard embedded software when/how this call should be used.

Clearing of keys that are provided by the smartcard embedded software to the Crypto Library is the responsibility of the smartcard embedded software.

This SFR depends on the availability of the N7121 Crypto Library.

6.1.4.4 Library Support for Elliptic Curve Cryptography (optional)

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

FCS_COP.1.1/ECDSA The TSF shall perform signature generation and verification in accordance with a specified cryptographic algorithm ECDSA / ECC over GF(p) and cryptographic key sizes 224, 256, 320, 384, 512 and 521 bits that meets the following: [ISO/IEC 14888-3], [ANSI X9.62-2005],[RFC 5639], [ANSSI 2011],[FIPS 186-4] and [IEEE Std 1363].

Application Note: The security functionality is resistant against side channel analysis and other attacks described in [JIL-ATT-SC]. To fend off attackers with high attack potential an adequate key length must be used (references can be found in national and international documents and standards).

Due to BSI regulations the certification covers the standard curves ansix9p224r1, ansix9p256r1, ansix9p384r1 and ansix9p521r1 from ANSI X9.62 [ANSI X9.62-1999], brainpoolP224r1, brainpoolP224t1, brainpoolP256r1, brainpoolP256t1, brainpoolP320r1, brainpoolP320t1, brainpoolP384r1, brainpoolP384t1, brainpoolP512r1 and brainpoolP512t1 from RFC 5639 [RFC 5639] and ANSSI FRP256v1 [ANSSI 2011] curves.

This SFR depends on the availability of the N7121 Crypto Library.

FCS_COP.1/ECC_DHKE **Cryptographic operation – Diffie-Hellmann Key Exchange (optional)**

FCS_COP.1.1/ECC_DHKE The TSF shall perform Diffie-Hellmann Key Exchange in accordance with a specified cryptographic algorithm ECDSA / ECC over GF(p) and cryptographic key sizes 224, 256, 320, 384, 512 and 521 bits that meets the following: [ISO/IEC 11770-3], [ANSI X9.63], [RFC 5639], [ANSSI 2011] and [IEEE Std 1363].

Application Note: The security functionality is resistant against side channel analysis and other attacks described in [JIL-ATT-SC]. To fend off attackers with high attack potential an adequate key length must be used (references can be found in national and international documents and standards).

The security functionality does not provide the complete key exchange procedure, but only the point multiplication which is used for the multiplication of the private key with the communication partner’s public key. Therefore this function can be used as part of a Diffie-Hellman key exchange as well pure point multiplication.

Due to BSI regulations the certification covers the standard curves ansix9p224r1, ansix9p256r1, ansix9p384r1 and ansix9p521r1 from ANSI X9.62 [ANSI X9.62-1999], brainpoolP224r1, brainpoolP224t1, brainpoolP256r1, brainpoolP256t1, brainpoolP320r1, brainpoolP320t1, brainpoolP384r1, brainpoolP384t1, brainpoolP512r1 and brainpoolP512t1 from RFC 5639 [RFC 5639] and ANSSI FRP256v1 [ANSSI 2011] curves.

This SFR depends on the availability of the N7121 Crypto Library.

FCS_CKM.1/ECDSA

Cryptographic Key Generation – ECDSA (optional)

FCS_CKM.1.1/ECDSA

The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm ECDSA (ECC over GF(p)) and specified cryptographic key sizes 224, 256, 320, 384, 512 and 521 bits that meet the following: [ISO/IEC 14888-3], [ANSI X9.62-2005] and [FIPS 186-4].

Application Note:

The security functionality is resistant against side channel analysis and other attacks described in [JIL-ATT-SC]. To fend off attackers with high attack potential an adequate key length must be used (references can be found in national and international documents and standards).

Due to BSI regulations the certification covers the standard curves ansix9p224r1, ansix9p256r1, ansix9p384r1 and ansix9p521r1 from ANSI X9.62 [ANSI X9.62-1999], brainpoolP224r1, brainpoolP224t1, brainpoolP256r1, brainpoolP256t1, brainpoolP320r1, brainpoolP320t1, brainpoolP384r1, brainpoolP384t1, brainpoolP512r1 and brainpoolP512t1 from RFC 5639 [RFC 5639] and ANSSI FRP256v1 [ANSSI 2011] curves.

This SFR depends on the availability of the N7121 Crypto Library.

FCS_CKM.4/ECDSA

Cryptographic Key Destruction – ECDSA (optional)

FCS_CKM.4.1/ECDSA

The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method overwrite that meets the following: [ISO 11568-4].

Application Note:

The crypto library provides the smartcard embedded software with library calls to perform various cryptographic algorithms that involve keys. Through the parameters of the library calls, the smartcard embedded software provides keys for the cryptographic algorithms. To perform its cryptographic algorithms, the library copies these keys, or a transformation thereof, to the working-buffer (supplied by the smartcard embedded software) and/or the memory/special function registers of the Crypto Library. Depending upon the algorithm the library either overwrites these keys before returning control to the smartcard embedded software or provides a library call to through which the smartcard embedded software can clear these keys. In the case of a separate library call to clear keys the guidance instructs the smartcard embedded software when/how this call should be used.

Clearing of keys that are provided by the smartcard embedded software to the Crypto Library is the responsibility of the smartcard embedded software.

This SFR depends on the availability of the N7121 Crypto Library.

6.1.4.5 Library Support for Hashing (optional)

Package "Hashing" defined in [PP].

FCS_COP.1/SHA **Cryptographic operation – Hashing (optional)**

FCS_COP.1.1/SHA The TSF shall perform hashing in accordance with a specified cryptographic algorithm SHA-1, SHA-224, SHA-256, SHA-384, and SHA-512 and cryptographic key sizes none that meets the following: [FIPS 180-4].

6.1.5 Further Security Functional Requirements – PUF (optional)

FCS_COP.1/AES_PUF **Cryptographic operation – PUF based AES**

FCS_COP.1.1/AES_PUF The TSF shall perform decryption and encryption in accordance with a specified cryptographic algorithm AES in CBC mode and cryptographic key sizes 128 bits that meets the following: [FIPS 197], [NIST SP 800-38A].

FCS_COP.1/MAC_PUF **Cryptographic operation – PUF based MAC**

FCS_COP.1.1/MAC_PUF The TSF shall perform MAC generation and calculation of CBC-MAC values used for PUF authentication in accordance with a specified cryptographic algorithm AES in CBC-MAC mode and cryptographic key sizes 128 bits that meets the following: [FIPS 197] and [ISO/IEC 9797-1] (MAC algorithm 1).

FCS_CKM.1/PUF **Cryptographic Key Generation – PUF**

FCS_CKM.1.1/PUF The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm key derivation function based on PUF and specified cryptographic key sizes 128 bits that meet the following: *PUF Key derivation function specification*, NXP Semiconductors, BUID, 2014. [PUF].

FCS_CKM.4/PUF **Cryptographic Key Destruction – PUF**

FCS_CKM.4.1/PUF The TSF shall destroy cryptographic keys **derived by key derivation function based on PUF** in accordance with a specified cryptographic key destruction method flushing of key registers that meets the following: none.

6.1.6 Further Security Functional Requirements – Self-tests

FPT_TST.1 **TSF Testing**

FPT_TST.1.1 The TSF shall run a suite of self tests *at the request of the authorised user* to demonstrate the correct operation of *the active shielding and the sensors*.

FPT_TST.1.2 The TSF shall provide authorised users with the capability to verify the integrity of *Special Function Registers holding static values loaded during start-up*.

FPT_TST.1.3 The TSF shall provide authorised users with the capability to verify the integrity of *stored TSF executable code*.

Application Note: In conformance with [CC_Part2], the TSF testing only addresses parts of the TSF. Therefore, the operations performed are selections and assignments which is indicated as *underlined italic text*. This is in conformance with the notation defined in Section 6.

This SFR is in place for all configurations of the TOE.

6.1.7 Further Security Functional Requirements – Management Functions

FMT_SMF.1 **Specification of Management Functions**

FMT_SMF.1.1 The TSF shall be capable of performing the following management functions:

- change of TOE mode to lower privileged mode by calling one of the following instructions: User Call,

- change of TOE mode to higher privileged mode by calling one of the following instructions: System Call,
- change of TOE mode by invoking an interrupt, and
- change of TOE mode by finishing an interrupt.

Application Note: This SFR is in place for all configurations of the TOE.

6.1.8 Further Security Functional Requirements – Access Control Policy

The Access Control Policy defines rules for accessing memory segments, and peripherals. Access control definitions for memory segments are stored in segment descriptors in RAM and fetched by the Segment Lookaside Buffer (SLB). Peripheral access information is stored in the Peripheral Access Control (PAC) Special Function Registers.

FDP_ACC.1/ACP **Subset Access Control – Access Control Policy**

FDP_ACC.1.1/ACP The TSF shall enforce the Access Control Policy on all code running on the TOE, all memories and all memory operations.

Application Note: The Access Control Policy shall be enforced by implementing a Memory Management Unit, which implements access control via the mapping of virtual memory addresses to physical memory addresses. The CPU always uses virtual memory addresses, which are mapped to physical memory addresses by the Memory Management Unit. Prior to accessing the respective memory address, the Memory Management Unit checks if the access is allowed. A denied read or write access is treated as a security violation and will trigger a Security Reset. This applies for all memories, including the Special Function Registers. The attempt to access a code segment for which no segment descriptor exists causes a "code miss" exception.

This SFR is in place for all configurations of the TOE.

FDP_ACF.1/ACP **Security Attribute Based Access Control – Access Control Policy**

FDP_ACF.1.1/ACP The TSF shall enforce the Access Control Policy to objects based on the following: segment descriptors and peripheral access information.

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

- code running in a certain mode is allowed to access data and/or execute code of data segments and/or from code segments of this mode,

- code running in Super-System Mode (which also comprises Test Mode and Configuration Mode) as well as code running in System Mode of Card A or B is allowed to read card configurations stored in Special Function Registers,
- code running in Test Mode and Configuration Mode is allowed to write card configurations stored in Special Function Registers,
- code running in any mode is allowed to read peripheral access information stored in Special Function Registers,
- code running in Super-System Mode (which also comprises Test Mode and Configuration Mode) as well as System Mode of Card A or B is allowed to write peripheral access information stored in Special Function Registers, and
- code running in Configuration Mode and Test Mode is allowed to read and write self-test related content stored in Special Function Registers.

FDP_ACF.1.3/ACP

The TSF shall explicitly authorize access of subjects to objects based on the following additional rules:

- code running in a certain mode is allowed to access data and/or execute code of another mode if this other mode explicitly shares the data and/or code with the currently running mode,
- code running in any mode is allowed to access cryptographic coprocessors, the RNG, the communication interface and the timer if the respective mode owns the requested hardware resource.

FDP_ACF.1.4/ACP

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

Application Note:

This SFR is in place for all configurations of the TOE.

There is a strict seperation between Card A and Card B for System Mode and User Mode. In context of the Secure UM Box in Card A, NXP defines access rights for the code running in UM of Card A. The NXP SM OS running in SM of Card A does not provide an interface to the UM application to change these values.

Note that card configurations for ROM stored in Special Function Registers ROM_FW_CRD_A and ROM_FW_REF_A are not writeable and card configurations stored in Special Function Registers RAM_FW_CRD_A and WL_TPP_SZ allow additional access, see [DSheet_MMU].

FMT_MSA.1/ACP

Management of Security Attributes – Access Control Policy

FMT_MSA.1.1/ACP

The TSF shall enforce the Access Control Policy to restrict the ability to *modify* the security attributes segment descriptors and peripheral access information to code executed in a TOE mode which has write access to the respective segment or Special Function Registers.

Application Note:

This SFR is in place for all configurations of the TOE.

FMT_MSA.3/ACP

Static Attribute Initialization – Access Control Policy

FMT_MSA.3.1/ACP

The TSF shall enforce the Access Control Policy to provide *restrictive* default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2/ACP

The TSF shall allow $_{\text{the}}$ no subject to specify alternative initial values to override the default values when an object or information is created.

Application Note:

The ST applies a further refinement by deleting a single word. This refinement is indicated. This refinement does not affect the meaning of the SFR and was applied for grammatical reasons only.

Restrictive means that the reset values of the peripheral access Special Function Registers are set to Super System Mode. In context of the Secure UM Box in Card A, the Access Control Policy implements NXP-defined restrictive default values for segment descriptors.

The TOE does not provide objects or information that can be created, since it provides access to memory areas. The definition of objects that are stored in the TOE's memory is subject to the Security IC Embedded Software.

This SFR is in place for all configurations of the TOE.

6.2 Security assurance requirements

Table 14 lists all security assurance requirements that are valid for this Security Target. These security assurance requirements are defined in [PP] and/or in [CC_Part3] for EAL6, except for requirements ASE_TSS.2 and ALC_FLR.1, which are augmentations of this Security Target to EAL6, see Section 2.2. ASE_TSS.2 is an augmentation in this Security Target to give architectural information on the security functionality of the TOE. ALC_FLR.1 is an augmentation in this Security Target to cover policies and procedures of the developer applied to track and correct flaws and support surveillance of the TOE.

In compliance with Application Note 22 in [PP] the third column in Table 14 shows, which Security Assurance Requirements (SARs) are added to this Security Target compared to [PP]. In this context, entry ”EAL6 / PP” means, that the requirement is defined in both, [CC_Part3] for EAL6 and [PP], entry ”EAL6” means, that the requirement is defined in [CC_Part3] for EAL6 but not in [PP], and entry ”ST” means, that the requirement is defined neither in [CC_Part3] for EAL6 nor in [PP], but in this Security Target.

All refinements of the security assurance requirements in the [PP], which must be adapted for EAL6, are described in this section.

Table 14. SARs for this ST

SAR	Title	Required by
ADV_ARC.1	Security architecture description	EAL6 / PP
ADV_FSP.5	Complete semi-formal functional specification with additional error information	EAL6
ADV_IMP.2	Complete mapping of the implementation representation of the TSF	EAL6
ADV_INT.3	Minimally complex internals	EAL6
ADV_TDS.5	Complete semiformal modular design	EAL6
ADV_SPM.1	Formal TOE security policy model	EAL6
AGD_OPE.1	Operational user guidance	EAL6 / PP
AGD_PRE.1	Preparative procedures	EAL6 / PP
ALC_CMC.5	Advanced support	EAL6
ALC_CMS.5	Development tools CM coverage	EAL6
ALC_DEL.1	Delivery procedures	EAL6 / PP
ALC_DVS.2	Sufficiency of security measures	EAL6 / PP
ALC_FLR.1	Basic flaw remediation	ST
ALC_LCD.1	Developer defined life-cycle model	EAL6 / PP
ALC_TAT.3	Compliance with implementation standards – all parts	EAL6
ASE_CCL.1	Conformance claims	EAL6 / PP
ASE_ECD.1	Extended components definition	EAL6 / PP
ASE_INT.1	ST introduction	EAL6 / PP
ASE_OBJ.2	Security objectives	EAL6 / PP
ASE_REQ.2	Derived security requirements	EAL6 / PP
ASE_SPD.1	Security problem definition	EAL6 / PP
ASE_TSS.2	TOE summary specification with architectural design summary	ST
ATE_COV.3	Rigorous analysis of coverage	EAL6
ATE_DPT.3	Testing: modular design	EAL6
ATE_FUN.2	Ordered functional testing	EAL6
ATE_IND.2	Independent testing – sample	EAL6 / PP
AVA_VAN.5	Advanced methodical vulnerability analysis	EAL6 / PP

In the set of assurance components chosen for EAL6, the assignment appears only in ADV_SPM.1. The assignment for ADV_SPM.1 is defined below.

ADV_SPM.1 Formal TOE security policy model

ADV_SPM.1.1D

The developer shall provide a formal security policy model for the

- Limited Capability and Availability Policy,
- Access Control Policy, and
- Loader SFP.

ADV_SPM.1.2D

For each policy covered by the formal security policy model, the model shall identify the relevant portions of the statement of SFRs that make up that policy.

ADV_SPM.1.3D

The developer shall provide a formal proof of correspondence between the model and any formal functional specification.

ADV_SPM.1.4D

The developer shall provide a demonstration of correspondence between the model and the functional specification.

6.3 Security requirements rationale

6.3.1 Rationale for the Security Functional Requirements

Section 6.3.1 in [PP] provides a rationale for the mapping between security functional requirements and security objectives defined in [PP]. The mapping is reproduced in the following table. Notice, that only TOE objectives are listed since no SFRs are mapped to objectives related to operational resp. development environment.

Table 15. Subjects, objects as well as related operations and attributes of the Loader Policy

Objective	SFR	Rationale
Objectives of the Protection Profile [PP]		
O.Leak-Inherent	FDP_ITT.1	See [PP].
	FDP_IFC.1	See [PP].
	FPT_ITT.1	See [PP].
O.Phys-Probing	FDP_SDC.1	See [PP].
	FPT_PHP.3	See [PP].
O.Malfunction	FPT_FLS.1	See [PP].
	FRU_FLT.2	See [PP].
O.Phys-Manipulation	FDP_SDI.2	See [PP].
	FPT_PHP.3	See [PP].
O.Leak-Forced	FDP_ITT.1	See [PP].
	FDP_IFC.1	See [PP].
	FPT_FLS.1	See [PP].
	FPT_ITT.1	See [PP].
	FPT_PHP.3	See [PP].

Objective	SFR	Rationale
	FRU_FLT.2	See [PP].
O.Abuse-Func	FDP_ITT.1	See [PP].
	FDP_IFC.1	See [PP].
	FMT_LIM.1	See [PP].
	FMT_LIM.2	See [PP].
	FPT_FLS.1	See [PP].
	FPT_ITT.1	See [PP].
	FPT_PHP.3	See [PP].
	FRU_FLT.2	See [PP].
O.Identification	FAU_SAS.1	See [PP].
O.RND	FCS_RNG.1/PTG.2	See [PP].
	FDP_ITT.1	See [PP].
	FDP_IFC.1	See [PP].
	FPT_FLS.1	See [PP].
	FPT_ITT.1	See [PP].
	FPT_PHP.3	See [PP].
	FRU_FLT.2	See [PP].
	FCS_RNG.1/PTG.3	The PTG.3 random number generator provided by the crypto library (if available) corresponds to a hybrid-physical random number generator. As an alternative to the physical true random number generator, this RNG corresponds to the objective O.RND.
	FCS_RNG.1/DRG.4	The DRG.4 random number generator provided by the crypto library (if available) corresponds to a hybrid-deterministic random number generator. As an alternative to the physical true random number generator, this RNG corresponds to the objective O.RND.
O.Cap_Avail_Loader	FMT_LIM.1/Loader	See [PP].
	FMT_LIM.2/Loader	See [PP].
O.Ctrl_Auth_Loader	FDP_ACC.1/Loader	See [PP].
	FDP_ACF.1/Loader	See [PP].
	FDP_UCT.1/Loader	See [PP].
	DP_UIT.1/Loader	See [PP].
	FTP_ITC.1/Loader	See [PP].
O.TDES	FCS_COP.1/TDES	See [PP].
	FCS_CKM.4/TDES	See [PP].

Objective	SFR	Rationale
	FCS_COP.1/TDES_LIB	See [PP].
	FCS_CKM.4/TDES_LIB	See [PP].
O.AES	FCS_COP.1/AES	See [PP].
	FCS_CKM.4/AES	See [PP].
	FCS_COP.1/AES_LIB	See [PP].
	FCS_CKM.4/AES_LIB	See [PP].
O.SHA	FCS_COP.1/SHA	See [PP].
Additional objectives defined in this ST		
O.NVM-Integrity	FDP_SDI.2	The objective requires integrity protection and correction of failures of data stored in the Flash memory.The SFR describes this functionality for all memories.
O.Access-Control O.Secure-User-Mode-Box (optional)	FDP_ACC.1/ACP FMT_MSA.1/ACP FMT_MSA.3/ACP FDP_ACF.1/ACP FMT_SMF.1	The objective requires access control to memories and special function registers. This is covered by the SFRs which define the access control policy. The SFRs also cover the additional and optional functionality of the NXP Secure User Mode Box defined by O.Secure-User-Mode-Box. The SFR FMT_SMF.1 requires functionality to change the TOE mode in a controlled way using User and System Calls or via interrupts triggered by hardware peripherals as described in O.Access-Control. The Access Control Policy is based on these TOE modes.
O.Self-Test	FPT_TST.1	The objective described self-test functionality to detect physical manipulation. The SFR FPT_TST.1 addresses the objective as it requires tests of the active shielding and sensors, integrity check of special function registers on start-up, and integrity check of stored TSF executable code.
O.PUF	FCS_COP.1/AES_PUF FCS_COP.1/MAC_PUF FCS_CKM.1/PUF FCS_CKM.4/PUF	The objective describes sealing and unsealing of user data using the device-specific PUF. This comprises encryption/decryption (FCS_COP.1/AES_PUF) and MAC calculation (FCS_COP.1/MAC_PUF). The SFRs FCS_CKM.1/PUF and FCS_CKM.4/PUF describe the generation and destruction of the device specific PUF key during start-up and shut down, respectively.

Objective	SFR	Rationale
O.RSA	FCS_COP.1/RSA FCS_COP.1/RSA_PAD FCS_COP.1/RSA_PubExp FCS_CKM.1/RSA FCS_CKM.4/RSA	The SFRs directly implement the functionality required by the objective.
O.ECC	FCS_COP.1/ECDSA FCS_COP.1/ECC_DHKE FCS_CKM.1/ECDSA FCS_CKM.4/ECDSA	The SFRs directly implement the functionality required by the objective.

6.3.2 Dependencies of the Security Functional Requirements

The dependencies listed in [PP] are independent of the additional dependencies listed in the table below.

The dependencies of the PP are fulfilled within the PP and at least one dependency is considered to be satisfied.

The following discussion demonstrates how the dependencies defined by Part 2 of the Common Criteria [CC_Part2] for the requirements specified in Section 6.1 and Section 6.2 are satisfied.

The dependencies defined in the Common Criteria are listed in the table below.

Table 16. Subjects, objects as well as related operations and attributes of the Loader Policy

SFR	Dependencies	Fulfilled by
SFRs of the Protection Profile		
FRU_FLT.2	FPT_FLS.1	See [PP].
FPT_FLS.1	--	See [CC_Part2].
FMT_LIM.1	FMT_LIM.2	See [PP].
FMT_LIM.2	FMT_LIM.1	See [PP].
FAU_SAS.1	--	See [PP].
FDP_SDC.1	--	See [PP].
FDP_SDI.2	--	See [CC_Part2].
FPT_PHP.3	--	See [CC_Part2].
FDP_ITT.1	FDP_ACC.1 or FDP_IFC.1	See [PP].
FPT_ITT.1	--	See [CC_Part2].
FDP_IFC.1	FDP_IFF.1	See [PP].
FCS_RNG.1/PTG.2	--	See [PP].
SFRs of the Loader Packages of the Protection Profile		
FMT_LIM.1/Loader	FMT_LIM.2	See [PP].
FMT_LIM.2/Loader	FMT_LIM.1	See [PP].
FTP_ITC.1/Loader	--	See [CC_Part2].
FDP_UCT.1/Loader	FTP_ITC.1 or FTP_TRP.1 FDP_ACC.1 or FDP_IFC.1	See [PP].

SFR	Dependencies	Fulfilled by
FDP_UIT.1/Loader	FDP_ACC.1 or FDP_IFC.1 FPT_ITC.1 or FTP_TRP.1	See [PP].
FDP_ACC.1/Loader	FDP_ACF.1	See [PP].
FDP_ACF.1/Loader	FDP_ACC.1 FMT_MSA.3	See [PP].
SFRs of the package “TDES” defined in the Protection Profile (partly optional)		
FCS_COP.1/TDES	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 FCS_CKM.4	See [PP]. FCS_CKM.1 is not covered by the ST. Key generation has to be provided by the Security IC Embedded Software.
FCS_COP.1/TDES_LIB (optional)	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 FCS_CKM.4	Partly fulfilled by FCS_CKM.4/TDES_LIB. FCS_CKM.1 is not covered by the ST. Key generation has to be provided by the Security IC Embedded Software.
FCS_CKM.4/TDES	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	See [PP]. FCS_CKM.1 is not covered by the ST. Key generation has to be provided by the Security IC Embedded Software.
FCS_CKM.4/TDES_LIB (optional)	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	See FCS_CKM.4/TDES. FCS_CKM.1 is not covered by the ST. Key generation has to be provided by the Security IC Embedded Software.
SFRs of the package “AES” defined in the Protection Profile (partly optional)		
FCS_COP.1/AES	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 FCS_CKM.4	See [PP]. FCS_CKM.1 is not covered by the ST. Key generation has to be provided by the Security IC Embedded Software.
FCS_COP.1/AES_LIB (optional)	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 FCS_CKM.4	Partly fulfilled by FCS_CKM.4/AES_LIB. FCS_CKM.1 is not covered by the ST. Key generation has to be provided by the Security IC Embedded Software.
FCS_CKM.4/AES	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	See [PP]. FCS_CKM.1 is not covered by the ST. Key generation has to be provided by the Security IC Embedded Software.
FCS_CKM.4/AES_LIB (optional)	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	See FCS_CKM.4/AES. FCS_CKM.1 is not covered by the ST. Key generation has to be provided by the Security IC Embedded Software.
SFRs from the optional package "Hashing" defined in the Protection Profile		

SFR	Dependencies	Fulfilled by
FCS_COP.1/SHA	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 FCS_CKM.4	See [PP]. As no key is used, there is no need for key import as required by dependency to FDP_ITC.1 or FDP_ITC.2 or key generation as required by FCS_CKM.1 or key destruction as required by the dependency to FCS_CKM.4. Therefore, there is no need to fulfill the dependencies.
Further SFRs defined in this ST		
FCS_COP.1/AES_PUF	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 FCS_CKM.4	Fulfilled by FCS_CKM.1/PUF and FCS_CKM.4/PUF.
FCS_COP.1/MAC_PUF	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 FCS_CKM.4	Fulfilled by FCS_CKM.1/PUF and FCS_CKM.4/PUF.
FCS_CKM.1/PUF	FCS_CKM.2 or FCS_COP.1 FCS_CKM.4	Fulfilled by FCS_COP.1/AES_PUF, FCS_COP.1/MAC_PUF and FCS_CKM.4/PUF.
FCS_CKM.4/PUF	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	Fulfilled by FCS_CKM.1/PUF.
FPT_TST.1	--	See [CC_Part2].
FMT_SMF.1	--	See [CC_Part2].
FDP_ACC.1/ACP	FDP_ACF.1	Fulfilled by FDP_ACF.1/ACP.
FDP_ACF.1/ACP	FDP_ACC.1 FMT_MSA.3	Fulfilled by FDP_ACC.1/ACP and FMT_MSA.3/ACP.
FMT_MSA.1/ACP	FDP_ACC.1 or FDP_IFC.1, FMT_SMR.1 FMT_SMF.1	Partly fulfilled by FDP_ACC.1/ACP and FMT_SMF.1. FMT_SMR.1: See discussion below.
FMT_MSA.3/ACP	FMT_MSA.1 FMT_SMR.1	Partly fulfilled by FMT_MSA.1/ACP. FMT_SMR.1: See discussion below.
FCS_RNG.1/DRG.4	--	See [PP].
FCS_RNG.1/PTG.3	--	See [PP].
FCS_COP.1/RSA	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 FCS_CKM.4	Fulfilled by FCS_CKM.1/RSA and FCS_CKM.4/RSA.
FCS_COP.1/RSA_PAD	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 FCS_CKM.4	Fulfilled by FCS_CKM.1/RSA and FCS_CKM.4/RSA.
FCS_COP.1/RSA_PubExp	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 FCS_CKM.4	Fulfilled by FCS_CKM.1/RSA and FCS_CKM.4/RSA.
FCS_COP.1/ECDSA	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 FCS_CKM.4	Fulfilled by FCS_CKM.1/ECDSA and FCS_CKM.4/ECDSA.

SFR	Dependencies	Fulfilled by
FCS_COP.1/ECC_DHKE	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 FCS_CKM.4	Fulfilled by FCS_CKM.1/ECDSA and FCS_CKM.4/ECDSA.
FCS_CKM.1/RSA	FCS_CKM.2 or FCS_COP.1 FCS_CKM.4	Fulfilled by FCS_COP.1/RSA and FCS_CKM.4/RSA.
FCS_CKM.1/ECDSA	FCS_CKM.2 or FCS_COP.1 FCS_CKM.4	Fulfilled by FCS_COP.1/ECDSA and FCS_CKM.4/ECDSA.
FCS_CKM.4/RSA	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	Fulfilled by FCS_CKM.1/RSA.
FCS_CKM.4/ECDSA	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	Fulfilled by FCS_CKM.1/ECDSA.

Note: *Please note that the partly fulfilled dependencies of the SFRs mapped to O.AES and O.TDES are given in case the crypto library is available or not. Even without the crypto library, the TSF provides functionality to destruct key as specified by FCS_CKM.4/AES and FCS_CKM.4/TDES, respectively. In any case, the Security IC Embedded Software has to implement key generation required by the missing dependency to FCS_CKM.1 of the hardware functionality and the functionality provided by the crypto library.*

Note: *The dependency to FMT_SMR.1 introduced by the components FMT_MSA.1/ACP and FMT_MSA.3/ACP is not applicable within the context of the SFRs. No additional definition of roles is required, as all necessary roles are already realized via the modes of the MMU. No actions by the Security IC Embedded Software Developer is required to implement those roles. In conclusion, these dependencies are not applicable.*

6.3.3 Refinements of the TOE Security Assurance Requirements

In compliance to Application Note 23 of [PP] this Security Target has to conform to all refinements of the security assurance requirements in [PP].These refinements are defined for the Security Assurance Requirements of EAL4. Thus, some of these refinements must be adapted to Security Assurance Requirements of higher levels according to EAL6 as claimed in this Security Target. All other Security Assurance Requirements defined in this Security Target and in particular the augmentations to EAL6 supplement and extent the Security Assurance Requirements in the [PP] and can be added without contradictions.

Table 17 lists all Security Assurance Requirements that are refined in the [PP] based on their definitions is [CC_Part3] and their effect on this Security Target.

Table 17. SARs refined in PP and their effect on this ST

Refined SAR in PP	Affected SAR in this ST	Rationale
ADV_ARC.1	ADV_ARC.1	SAR same as in [PP], refinement valid without change

Refined SAR in PP	Affected SAR in this ST	Rationale
ADV_FSP.4	ADV_FSP.5	The refinement in Section 6.2.1.6 of [PP] regarding ADV_FSP.4 addresses the complete representation of the TSF, the purpose and method of use of all TSFIs, and the accuracy and completeness of the SFR instantiations. The refinement is not a change in the wording of the action elements, but a more detailed definition of the items above. Compared to ADV_FSP.4 component ADV_FSP.5 requires a Functional Specification in a ”semi-formal style” (ADV_FSP.5.2C). In addition, component ADV_FSP.5 extends the scope of the error messages to be described from those resulting from an invocation of a TSFI (ADV_FSP.5.6C) to also those not resulting from an invocation of a TSFI (ADV_FSP.5.7C). For the latter a rationale shall be provided (ADV_FSP.5.8C). Since the higher level ADV_FSP.5 only affects the style of description and the scope of and rationale for error messages, the refinement in the [PP] regarding ADV_FSP.4 can be applied without changes and is valid for ADV_FSP.5.
ADV_IMP.1	ADV_IMP.2	The refinement in Section 6.2.1.7 of [PP] regarding ADV_IMP.1 states that it must be checked that the provided implementation representation is complete and sufficient to ensure that analysis activities are not curtailed due to lack of information. This Security Target targets assurance level EAL6 augmented, which requires access to all source code of the TOE so that the above refinement is implicitly fulfilled.
AGD_OPE.1	AGD_OPE.1	SAR same as in [PP], refinement valid without change.
AGD_PRE.1	AGD_PRE.1	SAR same as in [PP], refinement valid without change.
ALC_CMC.4	ALC_CMC.5	The refinement in Section 6.2.1.4 of [PP] regarding ALC_CMC.4 is a clarification of the ”TOE” and the term ”configuration items”. Since the higher level ALC_CMC.5 requires a higher assurance regarding the defined TOE and the configuration items, the refinement in [PP] regarding ADV_CMC.4 can be applied without changes and is valid for ADV_CMC.5.
ALC_CMS.4	ALC_CMS.5	The refinement in Section 6.2.1.3 of [PP] regarding ALC_CMS.4 is a clarification of the configuration item ”TOE implementation representation”. Compared to ALC_CMS.4 component ALC_CMS.5 only adds the requirement for a new configuration item to be included in the configuration list (ALC_CMS.51C) so that the refinement in the [PP] regarding ADV_CMS.4 can be applied without changes and is valid for ADV_CMS.5.
ALC_DEL.1	ALC_DEL.1	Same as in [PP], refinement valid without change.
ALC_DVS.2	ALC_DVS.2	Same as in [PP], refinement valid without change.

Refined SAR in PP	Affected SAR in this ST	Rationale
ATE_COV.2	ATE_COV.3	The refinement in Section 6.2.1.8 of [PP] regarding ATE_COV.2 defines that test coverage must include different operating conditions and ”ageing” and that existence and effectiveness of countermeasures against physical attacks cannot be tested but must be given by evidence. The refinement regarding test coverage is not a change in the wording of the action elements, but a more detailed definition of the items to be applied, so that it can be applied without changes and is valid for ATE_COV.3. The refinement regarding existence and effectiveness of countermeasures against physical attacks is implicitly fulfilled since this Security Target targets assurance level EAL6 augmented, which requires access to all source code and layout data
AVA_VAN.5	AVA_VAN.5	Same as in [PP], refinement valid without change. ***Note:*** *As required by Application Note 29 of the [PP], [JIL-ATT-SC] was utilized in its current version for the vulnerability analysis. The version is further more indicated in the reference list.*

6.3.4 Rationale for the Security Assurance Requirements

The selection of assurance components is based on the underlying [PP]. The Security Target uses the same augmentations as the PP (and the addition of augmentations ASE_TSS.2 and ALC_FLR.1), but chooses a higher assurance level. The level EAL6 is chosen in order to meet assurance expectations of digital signature applications and electronic payment systems. Additionally, the requirement of the PP to choose at least EAL4 is fulfilled.

The rationale for the PP augmentations is the same as in the PP. The assurance level EAL6 is an elaborated pre-defined level of the [CC_Part3]. The assurance components in an EAL level are chosen in a way that they build a mutually supportive and complete set of components. The requirements chosen for augmentation do not add any dependencies, which are not already fulfilled for the corresponding requirements contained in EAL6.

Therefore, these components add additional assurance to EAL6, but the mutual support of the requirements is still guaranteed.

As stated in the Section 6.3.3 of [PP], the TOE is intended to defend against sophisticated attacks. Therefore specifically AVA_VAN.5 was chosen by the PP in order to assure that even attackers with high attack potential cannot successfully attack the TOE.

In addition to the SARs introduced by EAL6, the following augmentations have been added:

- ASE_TSS.2 was chosen to include architectural information on the security functionality of the TOE in the ST.
- ALC_FLR.1 was chosen to prove that NXP tracks and corrects security flaws.

6.3.5 Security requirements are internally consistent

The discussion of security functional requirements and assurance components in the preceding sections has shown that mutual support and consistency are given for both groups of requirements. The arguments given for the fact that the assurance components are adequate for the functionality of the TOE also show that the security functional and assurance requirements support each other and that there are no inconsistencies between these groups.

The security functional requirements required to meet the security objectives O.Leak-Inherent, O.Phys-Probing, O.Malfunction, O.Phys-Manipulation and O.Leak-Forced also protect the cryptographic algorithms and the memory access/separation control function as well as the access control to Special Function Registers implemented according to the security functional requirements FDP_MSA.1/ACP, FDP_MSA.3/ACP and FDP_ACC.1/ACP, with reference to the Access Control Policy defined in FDP_ACF.1/ACP. Therefore, these Security Objectives support the secure implementation and operation of FDP_MSA.1/ACP, FDP_MSA.3/ACP and of FDP_ACC.1/ACP with FDP_ACF.1/ACP as well as the dependent security functional requirements.

A Security IC hardware platform requires Security IC Embedded Software to build a secure product. Thereby the Security IC Embedded Software must support the security functionality of the hardware and implement a sufficient management of the security services implemented in the hardware. The realization of the Security Functional Requirements within the TOE provides a good balance between flexible configuration and restrictions to ensure a secure behavior of the TOE.

7 TOE summary specification

7.1 Portions of the TOE Security Functionality

The TOE Security Functionality (TSF) directly corresponds to the TOE security functional requirements defined in Section 6.1 which are active and applicable to phases 4 to 7 of the Security IC product life-cycle defined in Section 1.2.3 of the [PP].

Note: *Please note that parts of the security functionality are configured at the end of phase 3 and all security functionality is active after phase 3 or phase 4 depending on the delivery form.*

Table 18. Portions of the TSF

TSF portion	Title	Description
TSF.Service	Service functionality beside cryptographic operations	This portion of the TSF comprises random number generation, reconfiguration of the TOE features, self-test functionality, as well as a secure channel for using the Flash Loader. It further provides mechanisms to store initialization, pre-personalization, and/or other data on the TOE.
TSF.Protection	General security measures to protect the TSF	This portion of the TSF comprises physical and logical protection to avoid information leakage and detect fault injection. It defines resets in case an error or attack was detected and guarantees that memories used by the optional available cryptographic libraries are cleared before other applications can access these memories.
TSF.Control	Operating conditions, memory and hardware access control	This portion of the TSF controls the operating conditions and manages the access rights to memories and peripherals for the different TOE modes.
TSF.Crypto	Crypto Service	This portion of the TSF provides cryptographic functionality such as TDES and AES in different modes depending on the availability of the N7121 Crypto Library. Furthermore, based on the availability of the N7121 Crypto Library, TSF.Crypto also covers asymmetric cryptography (RSA and ECC over GF(p)) and hashing.

7.2 TOE summary specification rationale

7.2.1 Mapping of Security Functional Requirements and TOE security functionality

The following table provides a mapping of portions of the TSF to SFRs. The table also provides information which aspect of the TSF covered by each SFR.

Table 19. Mapping of SFRs to portions of the TSF

SFR	TSF.Service	TSF.Protection	TSF.Control	TSF.Crypto	Description
SFRs of the Protection Profile					
FRU_FLT.2			X		Controls the operating conditions.
FPT_FLS.1		O	X		If the operating conditions are out of bounds, the TSF triggers a reset which restores a Secure State.
FMT_LIM.1			X		Blocking of test features after TOE delivery.
FMT_LIM.2			X		Blocking of test features after TOE delivery.
FAU_SAS.1	X				Store initialization data, pre-personalization data, and/or other data on the TOE
FDP_SDC.1		X			Memory data confidentiality.
FDP_SDI.2		X			Detects and counters integrity errors of data stored in memories.
FPT_PHP.3		X			Physical manipulation.
FDP_ITT.1		X			Information flow control to avoid information leakage.
FPT_ITT.1		X			Information flow control to avoid information leakage.
FDP_IFC.1		X			Information flow control to avoid information leakage.
FCS_RNG.1/PTG.2	X				Random number generation.
SFRs of the Loader Packages					
Loader Package 1 defined in the [PP]					
FMT_LIM.1/Loader			X		No disclosure of user data.
FMT_LIM.2/Loader			X		Block loader.
Loader Package 2 defined in the [PP] (optional)					
FTP_ITC.1/Loader	X	O			Trusted channel for using the loader.
FDP_UCT.1/Loader		X			Protect from unauthorized disclosure.
FDP_UIT.1/Loader		X			Protect from modification, deletion, insertion.
FDP_ACC.1/Loader			X		Defines on which Subjects and Objects the Loader SFP is applied.
FDP_ACF.1/Loader			X		Defines the Loader SFP.
SFRs of the Hardware Support for TDES and AES					
Package TDES defined in [PP]					
FCS_COP.1/TDES		O		X	TDES hardware support.
FCS_CKM.4/TDES		O		X	Destruction of cryptographic keys used by the TDES coprocessor.
Package "AES" defined in [PP]					
FCS_COP.1/AES		O		X	TDES hardware support.
FCS_CKM.4/AES		O		X	Destruction of cryptographic keys used by the AES coprocessor.
SFRs related to the crypto library (optional)					

SFR	TSF.Service	TSF.Protection	TSF.Control	TSF.Crypto	Description
Package Symmetric Ciphers (optional)					
FCS_COP.1/TDES_LIB		O		X	TDES library support.
FCS_COP.1/AES_LIB		O		X	AES library support.
FCS_CKM.4/TDES_LIB		O		X	Destruction of cryptographic keys used by the crypto library.
FCS_CKM.4/AES_LIB		O		X	
Package Random Number Generation (optional)					
FCS_RNG.1/DRG.4	X				Random number generation.
FCS_RNG.1/PTG.3	X				
Package RSA Encryption/Decryption and RSA Key Generation (optional)					
FCS_COP.1/RSA		O		X	RSA encryption, decryption, signature generation and verification.
FCS_COP.1/RSA_PAD		O		X	RSA message and signature encoding methods.
FCS_COP.1/RSA_ PubExp		O		X	RSA public key computation.
FCS_CKM.1/RSA				X	RSA key generation.
FCS_CKM.4/RSA		O		X	RSA key destruction.
Package ECC over GF(p) (optional)					
FCS_COP.1/ECDSA		O		X	ECDSA signature generation and verification.
FCS_COP.1/ECC_DHKE		O		X	Diffie-Hellmann Key Exchange via ECC over GF(p).
FCS_CKM.1/ECDSA				X	ECDSA key generation.
FCS_CKM.4/ECDSA		O		X	ECDSA key destruction.
SHA functionality defined in [PP] (optional)					
FCS_COP.1/SHA		O		X	Hashing with SHA-1, SHA-224, SHA-256, SHA-384, and SHA-512.
SFRs related to PUF					
FCS_COP.1/AES_PUF		O		X	AES-128 in CBC mode with PUF key.
FCS_COP.1/MAC_PUF		O		X	AES-128 in CBC-MAC mode with PUF key.
FCS_CKM.1/PUF				X	Key generation based on PUF.
FCS_CKM.4/PUF		O		X	Destruction of keys generated by PUF.
SFRs related to self-tests					
FPT_TST.1	X	O			Self-tests of TSF.
SFRs related to management functions					
FMT_SMF.1		X	O		Change of TOE modes via User Calls, System Calls and interrupts triggered by hardware peripherals.
SFRs related to the Access Control Policy					

SFR	TSF.Service	TSF.Protection	TSF.Control	TSF.Crypto	Description
FDP_ACC.1/ACP			X		Application of the ACP on Objects and Subjects.
FDP_ACF.1/ACP			X		Definition of the ACP.
FMT_MSA.1/ACP			X		Restrictive modification of ACP attributes of the defined objects by the defined subjects.
FMT_MSA.3/ACP			X		Restrictive default values for the ACP.

In the table above, ‘X’ indicates a direct mapping between SFR and portion of the TSF while ‘O’ indicates an indirect mapping

7.2.2 Security Architectural Information

Since this ST claims the assurance requirement ASE_TSS.2, security architectural information on a very high level is supposed to be included in the TSS to inform potential customers on how the TOE protects itself against interference, logical tampering and bypassing. In the security architecture context, this covers the aspects self-protection and non-bypassability. The aspects self-protection and non-bypassability are for a large part covered by TSF.Protection and TSF.Control. TSF.Protection covers the physical and logical protection of the TOE and protects the TOE against tampering and bypassing of security features and security services. It contributes by covering the aspects failure with preservation of a secure state and limited fault tolerance. This protects the TOE against interference of security feature and security services. TSF.Control limits the capability and availability of the Test Features and protects the TOE against bypassing of security features. In addition to the protection against interference, tampering and bypassing provided by TSF.Protection and TSF.Control, TSF.Service also contributes to the self-protection of the TOE and non-bypassing of security functionality.

7.2.2.1 TSF.Service

TSF.Service provides different functionality which are not directly related to cryptographic algorithms. It provides the following functionality:

TOE identification

FAU_SAS.1 is implemented by a test function that allows to store identification and/or pre-personalization data (including a unique ID for each die) for the TOE in the FLASH at the end of the tests in Phase 3. The FabKey Area as well as the configuration data stored in FLASH can be used by TSF.Service to store a unique identification for each die. Access to the FabKey area is limited by TSF.Control. Furthermore, this function is only available after a special authentication sequence. With regard to the unique ID for each die, TSF.Service can use the UID space in the configuration data segment or write data to the FabKey area that makes each die unique. Especially the FabKey Area depends on the choice of the Security IC Embedded Software developer and is included during ordering.

The FabKey area can be read out if TSF.Control grants access as required by FMT_SMF.1.

Trusted channel for usage of the Flash Loader (optional)

TSF.Service implements a secure communication protocol including mutual authentication, integrity protection and encryption. All protocols are based on AES-128 (CBC encryption, CMAC) and all data is transferred encrypted to provide confidentiality. Only integrity protected data are processed by the Flashloader OS and can be downloaded to the Flash. The Flash Loader establishes a secure channel for communication as required by FTP_ITC.1/Loader.

Self-test functionality

TSF.Service also provides a collection of functions that allows checking whether the TOE has been physically manipulated. These functions are also available to the Security IC Embedded Software to perform an on-demand self-test of the active shielding and sensors. TSF.Service implements exactly the functionality required by FPT_TST.1.

Note: *Please note that this does not comprise the self-tests available in the Chip-Health Mode (CHM) which are not defined as Security Functionality of the TOE.*

Hardware RNG following PTG.2 in [KS2011] of the German Common Criteria scheme

The physical RNG provides 8-bit random numbers. Within the allowed range of operating conditions (such as temperature), the RNG provides strong random numbers which fulfill the requirements of PTG.2 in [KS2011]. Only the power saving modes stops the random number generator. The physical RNG comprises a hardware test functionality to detect faults in the circuitry of the RNG (total failure test). Due to the fact that the hardware TOE does not support a Chi^2-test, it is up to the Security IC Embedded Software developer to implement the required online tests to detect whether low quality random numbers have been generated. Otherwise, the N7121 Crypto Library implements such a test which can be used by the Security IC Embedded Software if the library is available. Therefore TSF.Service partially meets FCS_RNG.1/PTG.2.

If the Security IC Embedded Software is implemented correctly the component protects itself against interference and allows the detection of tampering and bypassing.

Hybrid-deterministic RNG following DRG.4 in [KS2011] of the German Common Criteria scheme (optional)

The hybrid-deterministic RNG is implemented in the N7121 Crypto Library available for the TOE. The Library further provides the Security IC Embedded Software with an online test which can also be used to test the random numbers generated with the physical RNG to complete PTG.2 as required by FCS_RNG.1/PTG.2 if the user guidance is followed. The hybrid-deterministic random number generator according to [AIS31] DRG.4, meets FCS_RNG.1/DRG.4.

Hybrid-physical RNG following PTG.3 in [KS2011] of the German Common Criteria scheme (optional)

The hybrid-physical RNG is implemented in the Crypto Library available for the TOE. The Crypto Library further provides the Security IC Embedded Software with an online test which can also be used to test the random numbers generated with the physical RNG to complete PTG.2 as required by FCS_RNG.1/PTG.2 if the user guidance is followed. The hybrid-deterministic random number generator according to [AIS31] DRG.4, meets FCS_RNG.1/DRG.4.

7.2.2.2 TSF.Protection

TSF.Protection addresses functionalities of the TOE which are used to protect the TSF, TSF data and user data from any kind of attack. Its functionality mainly addresses self-protection of the TSF. However, TSF.Protection also addresses non-bypassability as it implements logical protection to avoid information leakage.

Integrity protection of memories

As required by FDP_SDI.2 , TSF.Protection supports the integrity of the ROM, RAM and Flash. The Flash is able to perform error correction. The ROM, RAM and Flash provide parity protection. A parity error is interpreted as a fault injection and forces a reset that increments the error counter. This combination increases the likelihood to detect manipulations of single cells. In addition, the manipulation of program or data or other meaningful values is much more difficult.

Furthermore, TSF.Protection also implements integrity protection during start-up. TSF.Protection supports all other SFRs because prevention of successful manipulation of security functionality is a pre-condition for the reliable work of all other functions.

Protection against physical manipulations

TSF.Protection protects the TOE against physical manipulation. In case a manipulation is detected, a reset is triggered to return to a secure state. Therefore, TSF.Protection implements FPT_PHP.3.

TSF.Protection comprises various special features in the design and layout of the circuitry, i.e., mainly shielding and hiding of relevant design parts. This includes

- security routing that adds unused lines between active ones to fill the topmost layers,
- route thick supply lines over interface areas,
- cover memory blocks and sensitive analogue parts with meshes and tiles, and
- using active lines (dummy lines with controlled signals) to cover important signal lines.

There is no common bus, only local dedicated data, code and address lines interconnect the different memories and the CPU. All interfaces, including the data, code and address encryption logic for the memories are part of the ’Glue Logic’. Therefore it is never possible to observe any clear data by tapping local memory buses.

Beside the measures mentioned above, the general CPU Functions, the hardware components, the memory management unit with all memory interfaces including the encryption functions are realized in so-called ’Glue Logic’. The glue logic is a sea of basic low level gates. These gates are placed and interconnected by using automated tools which provide random, heuristic and deterministic placement and routing procedures. The CPU bus is never leaving the ’Glue Logic’ area. The five metal layer technology allows routing on top of the cells. By this on one hand no routing channels can be found (and therefore also no bus structures can be found) and on the other hand even the cell structure itself is no longer visible.

These features also support all other SFRs because prevention of successful manipulation of security functionality is a pre-condition for the reliable work of all other functions.

As the self-test functionality required by FPT_TST.1 can be used to detect physical manipulations, it indirectly contributes to TSF.Protection. This corresponds to Application Note 20 of the PP.

This aspect of TSF.Protection is further supported by FPT_FLS.1 which controls the environmental conditions and triggers a reset in case these are out of bounds. Taking the environmental conditions to their specified boundaries might be part of an attack.

Logical protection

TSF.Protection prevents the reconstruction of TOE internal information that can be found by analysis of external measured signals like power or clock. Within the different components of the TOE dedicated functions are implemented to sufficiently limit or eliminate the information that might be contained in the shape and amplitude of signals or in the time between events.

The countermeasures implemented in the cryptographic co-processors (AES, TDES, and FAME3) such as blinding and randomization are independent of the keys and plain- or ciphertext calculated by the co-processor. The same calculation time for the encryption and decryption function (for a single operation) with all operands is also ensured by the design of the co-processor.

The clock configuration allows the usage of internally generated clock signals for different components (e.g. the co-processor) on the TOE to operate independent of the external clock. In addition the execution of instructions by the CPU is randomized to some extent to prevent the possibility to synchronize the internal behavior based on external signals (clock and power consumption) for leakage attacks. Security critical comparisons are protected by hardware and software countermeasures.

Other features like filtering and scrambling that are implemented to increase the robustness and confidentiality also contribute to counter leakage attacks.

Logical protections implemented by TSF.Protection covers the SFRs FDP_ITT.1, FPT_ITT.1 and FDP_IFC.1. They cannot be influenced from outside the TOE.

In addition, TSF.Protection encrypts contents stored in ROM, RAM and Flash memory with address-dependent keys and applies memory address scrambling. This ensures the confidentiality of user data stored in ROM, RAM and Flash memory as required by FDP_SDC.1.

Flash Loader data confidentiality and integrity protection (optional)

As already described for TSF.Service, the Flash Loader implements a secure channel for its usage. The implemented protocol also fulfills FDP_UCT.1/Loader and FDP_UIT.1/Loader which require that data transmitted via the trusted channel (FTP_ITC.1/Loader) is protected from unauthorized disclosure (data are transferred in encrypted form only) and integrity protected (MAC) to determine on receipt that if the transferred data has been modified.

The trusted channel required by FTP_ITC.1/Loader supports FDP_UCT.1/Loader and FDP_UIT.1/Loader and therefore indirectly supports TSF.Protection.

Cryptographic coprocessors and cryptographic library

The cryptographic coprocessors (TDES, AES and Fame3) as well as the cryptographic library implements countermeasures against fault injection and information leakage. For instance, these TOE components implement integrity protection of processed data. They further implement randomization such as blinding, dummy calculations and random delay before and after calculations. A futher implemented mechanism to protect User Data from unwanted disclosure is an automatic clean-up of relevant registers (key and data registers of the used coprocessor) after usage and before changing the TOE mode.

Therefore, all FCS_COP.1 and FCS_CKM.4 iterations indirectly support TSF.Protection.

The cryptographic library further implements the following features which are assigned to the data processing policy (FDP_ITT.1, FDP_ITT.1, and FDP_IFC.1) and FPT_PHP.3:

- The KeyStore feature can be used by the IC Embedded Software for a secure key-management in RAM. Keys are stored in an encrypted form. The KeyStore Manager uses AES-128 with a session-dependent master key for key encryption/decryption. It further stores key checksums in encrypted form.
- The crypto library further implements secure copy, move, and compare operations. These operations are protected against fault injection and information leakage.
- The crypto library implements basic support of the PACE protocol ([TR-03110-1], [TR-03110-2], [TR-03110-3], [TR-03110-4]) as ECC base-point operations are protected against fault injection and information leakage.

Protection of the general purpose I/O interface against misuse

The general purpose I/O interface is directly connected to the internal SFR bus. It can therefore be used to directly access peripherals of the TOE, such as the cryptographic coprocessors. In case this interface is not used, its externally accessible contact is mounted to ground. Therefore, no communication is possible from a functional perceptive. However, the TOE implements further measures to avoid misuse. It is not possible to observe TOE internal data via the interface as the SFR bus is mask protected. The TOE is further able to detect faults which might be induced via this interface as it implements internal integrity checks and protection.

7.2.2.3 TSF.Control

TSF.Control addresses those aspects the TSF controls, e.g., the operating conditions or access to specific memory addresses. Its functionality mainly addresses non-bypassability of the TSF.

Control of operating conditions

TSF.Control ensures the correct operation of the TOE hardware (functions offered by the micro-controller including the standard CPU, the crypto coprocessors, the memories, registers, I/O interfaces and the other system peripherals) during the execution of the IC Dedicated Support Software and Security IC Embedded Software. For this the TOE comprises filters for power supply and clock input. In addition, TSF.Control controls the allowed range of temperature, clock frequency, voltage and light.

The filters support the correct function of the TOE within the limits of the operating conditions. This robustness implements FRU_FLT.2 and ensures that the processing is performed without failure that may be caused by interference of any external communication interface or other external influences. Therefore the proper operation of the Random Number Generator and the cryptographic coprocessor that are used for cryptographic operations can be ensured within the specified limits. This also holds for the CPU and all other specialized components.

FPT_FLS.1 is implemented by sensors for the upper and lower threshold of the operating conditions temperature, clock frequency as well as voltage. The sensors detect whether one parameter is outside the specified range. Light sensors distributed over the chip surface detect abnormal light intensities. The secure state required by FPT_FLS.1 is realized by an internal reset of the Security IC. This secure state is applied as long as one sensor identifies an abnormal condition. Furthermore FPT_FLS.1 is also implemented by detecting fault injections in the cryptographic coprocessors, the CPU, memories and registers. Any detection of an attack will be signaled again by performing a reset which leads to the secure state. Access to a not implemented Special Function Register or memory address will also force a security reset. The Flash module also contains a light detection function. A light attack detected by the Flash also leads to a reset.

If one of the monitored parameters is out of the specified range, either a security reset is forced and execution is aborted or an interrupt is invoked that interrupts the program flow. Interrupts force a jump to a specific fixed address in the ROM or Flash. Any interrupt can therefore be controlled and guided by a specific part of the Security IC Embedded Software.

The TOE is equipped with a watchdog timer to protect the program execution flow against fault injection attacks. The watchdog timer needs to be enabled and configured by the Security IC Embedded Software. A time out of the watchdog timer leads to a security reset. The TOE further provides a code signature feature which can also be used for execution flow protection. Considering Application Note 14 of [PP], an internal reset of the Security IC is sufficient to ensure a secure state because all internal operations are stopped and the relevant special function registers are set to defined reset values. However, security mechanisms detecting faults, like on memories, cryptographic operations or CPU operations go beyond this requirement and increment the implemented error counter. The TOE distinguishes two severity levels of detected fault and limits the total accepted number of the more severe level. If this maximum is exceeded the Security IC Embedded Software will disable the TOE. Therefore the secure state forced by the hardware is extended by state in which the TOE is totally disabled if too many faults are detected. This is also a part of FPT_FLS.1.

The Security IC Embedded Software cannot disable the filters, sensors or any other kind of integrity protection. In addition the filters and sensors together with the reset block are implemented mostly independent of the other hardware components. This means that these parts of the TSF maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects and also by the Security IC Embedded Software. The features implemented by SF.OPC cannot be influenced from outside the TOE.

Mode control

TSF.Control realizes the control within the TOE testing phases (phase 3 of the life-cycle) and afterwards. The life-cycle ’Wafer Test’ is available for testing purposes in the phases before TOE delivery and disabled before the TOE is delivered from NXP to the customer.

The implemented control of the TOE mode ensures that in the Super System Mode the TOE

1. allows executing the IC Dedicated Test Software only in life-cycle ’Wafer Test’
2. allows executing the IC Dedicated Support Software
3. prevents from executing the Security IC Embedded Software.

In analogy it ensures that the TOE in the User Mode or System Mode

1. allows executing the Security IC Embedded Software in life-cycle ’Release’ and
2. prevents from executing the IC Dedicated Test Software.

TSF.Control provides access to the IC Dedicated Test Software in the Super System Mode before TOE delivery or to the IC Dedicated Support Software and Security IC Embedded Software after TOE delivery. The access is provided by evaluating the related electronic fuses during the boot sequence. It assures that it is not possible to enable access to the IC Dedicated Test Software after TOE delivery. Moreover it prevents direct access to the Special Function Registers.

In addition, TSF.Protection restricts the access for configuration of security features to the System Mode and partly to User Mode. This is supported by TSF.Control.

The configuration and trimming during life-cycle ’Wafer Test’ supports the correct operation. The correct configuration of the TOE during the boot sequence is supported by all security features. In this way the self-protection aspect and the protection from interference and tampering are implemented. The protection applies to all configuration values that are relevant.

The test concept with specific hardware operations initiated by the test software cannot be used to read out directly any data stored in one of the memories of the TOE. Therefore the capabilities to abuse the test functions for compromising User Data or TSF data is very limited as required by FMT_LIM.1.

At the end of the wafer test the access to the IC Dedicated Test Software is disabled. TSF.Control ensures that it is not possible to switch back and reuse the test functions again. In addition, the test functions of the IC Dedicated Test Software require a special sequence to execute a dedicated test routine. Therefore, TSF.Control limits the availability of the test functions as stated by FMT_LIM.2.

Access control to memories

TSF.Control ensures the access control policy by implementing the SFRs FDP_ACC.1/ACP and FDP_ACF.1/ACP and thereby providing different access rights for Super System Mode, System Mode and User Mode. Additionally, TSF.Control prevents access of the IC Dedicated Test Software by the Security IC Embedded Software. Therefore TSF.Control also realizes the SFR FMT_LIM.2.

The static attribute initialization FMT_MSA.3/ACP is given by the implementation of the hardware which enforces constant access rights to memory areas. There is no possibility to change the implemented access rights, therefore change is only possible if the TOE mode allows this, as defined by FMT_MSA.1/ACP. The only management function is the switch between Super System Mode, System Mode and User Mode as required by FMT_SMF.1. Due to the enforced hardware access control implemented in TSF.Control, the TOE protects itself against bypassing, which covers the aspect of non-bypassability in the security architecture context.

Access control to special function registers

TSF.Control realizes the access control to the Special Function Registers based on the TOE mode. Access to the Special Function Registers is granted or not depending on the TOE mode (Super System Mode, System Mode or User Mode). The access control is enforced by dividing the Special Function Registers into certain groups and enforcing the same access rights for all Special Function Registers belonging to a dedicated group. For each Special Function Register it is defined in which mode it can be accessed for reading and writing. Additionally, there is a peripheral access control concept which allows ownership of Special Function Registers.

Note that the control of the TOE mode is subject of TSF.Control. After testing of the TOE is completed at the end of phase 3, TSF.Protection prevents that the IC Dedicated Test Software can be executed. TSF.Control makes sure that Special Function Registers dedicated to test functionality are also not accessible after phase 3, therefore realizing FMT_LIM.2. If the Security IC Embedded Software tries to read or write an Special Function Register that is not implemented or where the access is denied, a reset is triggered which increments the error counter since this is seen as an attack. The implementation of TSF.Control realizes the SFRs FDP_ACC.1/ACP and FDP_ACF.1/ACP and thereby provides different access rights for Super System Mode, System Mode and User Mode.

The static attribute initialization FMT_MSA.3/ACP is given by the implementation of the hardware which enforces constant access rights to all Special Function Registers. There is no possibility to change the implemented access rights, therefore change is only possible if the TOE mode allows this, as defined by FMT_MSA.1/ACP and no management functions are available as specified by FMT_SMF.1.

TSF.Control also implements changing the TOE mode as specified by FMT_SMF.1. Changing the TOE mode can be accomplished by calling CPU instructions, or by invoking and finishing interrupts. Due to the enforced access rights, TSF.Control protects the TOE against bypassing, which covers the aspect of non-bypassability in the security architecture context.

Secure User Mode Box firewall

The NXP Secure User-Mode Box ensures that whatever user mode code is executed in UM of logical card A cannot endanger any asset of the TOE. This security feature allows to change the code running in User Mode of Card A without any impact on the TSF or the Security IC Embedded Software residing in Card B.

This feature is achieved as NXP defines restrictive values for segment descriptors. These restrictive values cannot be changed later on as the NXP System Mode OS running in SM of Card A does not provide interfaces to the UM to do that.

This functionality is covered by the Access Control Policy, i.e., FDP_ACC.1/ACP, FDP_ACF.1/ACP, FMT_MSA.1/ACP, and FMT_MSA.3/ACP.

Access control to Flash Loader functionality

TSF.Control implements access control for the usage of the Flash Loader. This comprises different user roles with different access rights. Furthermore, this comprises the definition of the Life Cycle State of the Flash Loader following the Loader Policy (see Section 6.1.2). Therefore, TSF.Control implements FDP_ACC.1/Loader and FDP_ACF.1/Loader.

Furthermore, TSF.Control also limits availability of the Flashloader OS according to FMT_LIM.1/Loader and FMT_LIM.2/Loader depending on the Life Cycle State of the Flash Loader. This applies to the permanent blocking of the flash loader after usage by the customer or blocking of the Flash Loader by NXP in case it is not selected via TOE configuration.

7.2.2.4 TSF.Crypto

TSF.Crypto covers the raw cryptographic functionality of the TSF (if available). Each of its components' availability depends on TOE configurations described in Table 1. TSF.Crypto does not address countermeasures against attacks or its internal use to avoid bypassability. These aspects are completely covered by TSF.Protection and TSF.Control, respectively. Therefore, TSF.Crypto does not address the aspects of self-protection or non-bypassability.

Hardware support for Triple-DES encryption/decryption

TSF.Crypto implements a coprocessor for Triple-DES operations. This coprocessor applies the encryption/decryption function to 16 bytes data. It provides an 8 byte key register that supports fast Triple-DES calculations. Therefore, TSF.Crypto is suitable to meet FCS_COP.1/TDES.

The coprocessor implements the Triple-DES algorithm in ECB mode as defined by [NIST SP 800-67] and [NIST SP 800-38A] by means of a hardware coprocessor and supports (a) the 3-key Triple-DES algorithm according to keying option 1 and (b) the 2-key Triple-

DES algorithm according to keying option 2 in [NIST SP 800-67]. The two/three 56-bit keys (112-/168-bit) for the 2-key/3-key Triple-DES algorithm shall be provided by the Security IC Embedded Software.

The Triple-DES coprocessor also supports hardware XOR-operation of two data blocks to support chaining modes of the TDES if this is configured by the Security IC Embedded Software.

The Triple-DES coprocessor further implements key destruction by flushing of key registers as required by FCS_CKM.4/TDES.

The power saving modes stops the operation of the Crypto2+ coprocessor.

Library support for Triple-DES encryption/decryption (optional)

If the crypto library is available, it implements further modes of operation on top of functionality provided by the coprocessor, thus meeting the application note of FCS_COP.1/TDES_LIB. The crypto library further implements cryptographic key destruction as required by FCS_CKM.4/TDES_LIB.

In addition to the ECB mode implemented in hardware, the N7121 Crypto Library implements ECB, CBC, CBC-MAC, Retail-MAC and CMAC mode following [NIST SP 800-67] (TDES) , [NIST SP 800-38A] (ECB and CBC mode) ,[ISO/IEC 9797-1] , Algorithm 1 (CBC-MAC mode), [ISO/IEC 9797-1], Algorithm 3 (Retail-MAC), and [NIST SP 800-38B] (CMAC mode) .

The resistance against SPA, DPA and timing attacks is supported by the hardware coprocessor. However, the N7121 Crypto Library implements additional countermeasures that are configurable at runtime and provides functionality for handling checksums over loaded keys.

Hardware support for AES encryption/decryption

TSF.Crypto implements a coprocessor for AES operations. This coprocessor applies the encryption/decryption function to 16 bytes data. It provides a key register supporting AES calculations with three different key sizes (128, 192 or 256 bit) following [FIPS 197] in ECB mode following [NIST SP 800-38A]. The AES is performed with a minimum control by the Security IC Embedded Software. The control of the AES within the encryption/decryption function is provided by an own sequencer of the coprocessor. Furthermore, the coprocessor implements flushing of key registers. The keys for the AES algorithm shall be provided by the Security IC Embedded Software.

The AES coprocessor also supports hardware XOR-operation of two data blocks to support chaining modes of the AES if this is configured by the Security IC Embedded Software.

Therefore, TSF.Crypto is suitable to meet FCS_COP.1/AES and FCS_CKM.4/AES.

The power saving modes stops the operation of the coprocessor.

Library support for AES encryption/decryption (optional)

If the crypto library is available, it implements further modes of operation on top of functionality provided by the coprocessor, thus meeting the application note of FCS_COP.1/AES_LIB. The N7121 Crypto Library further implements cryptographic key destruction as required by FCS_CKM.4/AES_LIB.

The TOE implements the AES following [FIPS 197] with different security configurations. The supported modes are ECB, ”outer” CBC and CTR following [NIST SP 800-38A] and CMAC (i.e. the CBC mode applied to the block cipher algorithm AES) following [NIST SP 800-38B]. In addition, the TOE provides the ability to compute a CBC-MAC following

[ISO/IEC 9797-1]. The CBC-MAC mode of operation is rather similar to the CBC mode of operation, but returns only the last cipher text.

The resistance against SPA, DPA and timing attacks is supported by the hardware coprocessor. However, the TOE implements additional countermeasures that are configurable at runtime and provides functionality for handling checksums over loaded keys.

PUF functionality

TSF.Crypto provides functionality to seal/unseal user data stored in shared memory. User data stored in shared memory can be encrypted/decrypted using the PUF block. A Message Authentication Code (MAC) can be calculated as a PUF authentication value. Hence, the user data can be sealed within the TOE and can be solely unsealed by the TOE.

The cryptographic key for sealing/unsealing of the user data is generated with the help of a key derivation function based on the PUF block and the Random Number Generator (RNG). The PUF block provides the PUF data to the key derivation function and thereby the cryptographic key is derived. If the TOE is powered off, the PUF data is not available from the PUF block. Also, derived keys from the key derivation function are unavailable, therefore this implements key destruction by flushing of key registers as required by FCS_CKM.4/PUF.

Therefore, TSF.Crypto is suitable to meet FCS_CKM.1/PUF and FCS_CKM.4/PUF.

The encryption/decryption of user data and the calculation of a MAC as a PUF authentication value are performed within the AES coprocessor by its own sequencer. Therefore, TSF.Crypto is suitable to meet FCS_COP.1/AES_PUF and FCS_COP.1/MAC_PUF. The power saving modes stop the operation of the PUF block

Library support for RSA (optional)

The N7121 Crypto Library contains modular exponentiation functions, which, together with other functions in the TOE, perform the operations required for RSA encryption and decryption. Two different RSA algorithms are supported by the TOE, namely the "Simple Straight Forward Method" (called RSA "straight forward", the key consists of the pair n and d) and RSA using the "Chinese Reminder Theorem" (RSA CRT, the key consists of the quintuple p, q, dp, dq, qInv). These algorithms are defined in [PKCS #1], v2.2 (RSAEP, RSADP, RSAP1, RSAVP1. This corresponds to the functionality required by FCS_COP.1/RSA.

The N7121 Crypto Library further provides functions that implement the RSA algorithm and RSA-CRT algorithm for message and signature encoding. This IT security functionality supports the EME-OAEP and EMSA-PSS signature scheme. All algorithms are defined in [PKCS #1], v2.2 (EME-OAEP, EMSA-PSS). This corresponds to the functionality required by FCS_COP.1/RSA_PAD.

Additionally, the N7121 Crypto Library provides functions that implement computation of an RSA public key from a private CRT key as defined in PKCS #1, v2.2. This corresponds to the functionality required by FCS_COP.1/RSA_PubExp.

Besides the derivation of public keys, the N7121 Crypto Library further supports generation of RSA key pairs as described in [PKCS #1], v2.2, [ALGO], and [FIPS 186-4]. This corresponds to the functionality required by FCS_CKM.1/RSA.

The crypto library also implements cryptographic key destruction as required by FCS_CKM.4/RSA.

TSF.Crypto supports RSA key lengths from 512 to 4096 bits.

Library support for ECC (optional)

The N7121 Crypto Library provides functions to perform ECDSA signature generation and signature verification according to [ISO/IEC 14888-3], [ANSI X9.62-2005], [FIPS 186-4] and [IEEE Std 1363]. Note that hashing of the message has to be done beforehand following FCS_COP.1/SHA. This corresponds to the functionality required by FCS_COP.1/ECDSA.

Furthermore, the N7121 Crypto Library provides functions to perform Diffie-Hellmann key exchange according to [ISO/IEC 11770-3], [ANSI X9.63] and [IEEE Std 1363]. This corresponds to the functionality required by FCS_COP.1/ECC_DHKE.

In addition, the N7121 Crypto Library also provides functions to perform ECC over GF(p) key generation according to [ISO/IEC 14888-3], [ANSI X9.62-2005] and [FIPS 186-4]. This functionality corresponds to FCS_CKM.1/ECDSA.

For some applications, a secret may be assigned to a specific point on the elliptic curve. This secret is then processed by means of curve arithmetics. The TSF provides secure point operations to protect these specific points to support the implementation of such applications.

The crypto library also implements cryptographic key destruction as required by FCS_CKM.4/ECDSA.

TSF.Crypto supports the following elliptic curves: ansix9p224r1, ansix9p256r1, ansix9p384r1 and ansix9p521r1 from ANSI X9.62 [ANSI X9.62-1999], brainpoolP224r1, brainpoolP224t1, brainpoolP256r1, brainpoolP256t1, brainpoolP320r1, brainpoolP320t1, brainpoolP384r1, brainpoolP384t1, brainpoolP512r1 and brainpoolP512t1 from RFC 5639 [RFC 5639] and ANSSI FRP256v1 [ANSSI 2011].

Library support for hashing (optional)

The N7121 Crypto Library provides functions to compute the Secure Hash Algorithms SHA-1, SHA-224, SHA-256, SHA-384, and SHA-512 according to [FIPS 180-4]. This corresponds to the functionality required by FCS_COP.1/SHA.

8 References

[AIS20] *Anwendungshinweise und Interpretationen zum Schema (AIS), Funktionalitätsklassen und Evaluationsmethodologie für deterministische Zufallszahlengeneratoren*, Version 3, 2013-05-15, Bundesamt für Sicherheit in der Informationstechnik.

[AIS26] *Anwendungshinweise und Interpretationen zum Schema (AIS), AIS 26, Evaluationsmethodologie für in Hardware integrierte Schaltungen*, Version 9, 2013-03-21, Bundesamt für Sicherheit in der Informationstechnik.

[AIS31] *Anwendungshinweise und Interpretationen zum Schema (AIS), AIS 31, Funktionalitätsklassen und Evaluationsmethodologie für physikalische Zufallszahlengeneratoren,* Version 3, 2013-05-15, Bundesamt für Sicherheit in der Informationstechnik.

[JIL-ATT-SC] *Joint Interpretation Library – Attack Methods for Smartcards and Similar Devices*, Version 2.2, 2013-01. Part of [AIS26].

[KS2011] *A proposal for: Functionality classes for random number generators*, W. Killmann, W. Schindler, Version 2.0, 2011-09-18, T-Systems GEI GmbH and Bundesamt für Sicherheit in der Informationstechnik. Part of [AIS20] and [AIS31].

[CC_Part1] *Common Criteria, Part 1: Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and general model*, Version 3.1, Revision 5, April 2017, CCMB-2017-04-001.

[CC_Part2] *Common Criteria, Part 2: Common Criteria for Information Technology Security Evaluation, Part 2: Security functional components*, Version 3.1, Revision 5, April 2017, CCMB-2017-04-002.

[CC_Part3] *Common Criteria, Part 3: Common Criteria for Information Technology Security Evaluation, Part 3: Security assurance components*, Version 3.1, Revision 5, April 2017, CCMB-2017-04-003.

[CEM] *Common Methodology for Information Technology Security Evaluation, Evaluation Methodology*, Version 3.1, Revision 5, April 2017, CCMB-2017-04-004.

[PP] *Security IC Platform Protection Profile with Augmentation Packages*, Version 1.0, registered and certified by Bundesamt für Sicherheit in der Informationstechnik (BSI) under the reference BSI-PP-0084-2014

[DSheet] *NXP Secure Smart Card Controller N7121 – Overview*, Product data sheet

[DSheet_InSet] *NXP Secure Smart Card Controller N7121 – Instruction Set Manual*, Objective data sheet addendum

[DSheet_CHM] *NXP Secure Smart Card Controller N7121 – Chip Health Mode*, Objective data sheet addendum

[DSheet_Periph] *NXP Secure Smart Card Controller N7121 – Peripheral Configuration and Use*, Objective data sheet addendum

[DSheet_MMU] *NXP Secure Smart Card Controller N7121 – MMU Configuration and NXP Firmware Interface Specification*, Objective data sheet addendum

[DSheet_FL] *NXP Secure Smart Card Controller N7121 – Flashloader OS*, Objective data sheet addendum

[DSheet_LibInt] *NXP Secure Smart Card Controller N7121 – Shared OS Libraries*, Objective data sheet addendum

[DSheet_SMOS] *NXP Secure Smart Card Controller N7121 – NXP System Mode OS*, Objective data sheet addendum

[UM_RNG] *N7121 Crypto Library – RNG Library*, Preliminary user manual

[UM_SymCfg] *N7121 Crypto Library – Symmetric Cipher Library (SymCfg)*, Preliminary user manual

[UM_KeyStore] *N7121 Crypto Library – KeyStoreMgr Library*, Preliminary user manual

[UM_SymUtils] *N7121 Crypto Library – Utils Library*, Preliminary user manual

[UM_RSA] *N7121 Crypto Library – RSA Library*, Preliminary user manual

[UM_RSAKeyGen] *N7121 Crypto Library – RSA Key Generation Library*, Preliminary user manual

[UM_ECC] *N7121 Crypto Library – ECC over GF(p) Library*,Preliminary user manual

[UM_SHA] *N7121 Crypto Library – SHA Library*, Preliminary user manual

[UM_HASH] *N7121 Crypto Library – HASH Library*, Preliminary user manual

[UM_AsymUtils] *N7121 Crypto Library – UtilsAsym Library*, Preliminary user manual

[GOM] *NXP Secure Smart Card Controller N7121, Information on Guidance and Operation*, Guidance and operation manual

[GOM_CL] *N7121 Crypto Library, Information on Guidance and Operation*, Product user manual

[PUF] *PUF Key derivation function specification*, NXP Semiconductors, BUID, 2014.

[ALGO] *Bekanntmachung zur elektronischen Signatur nach dem Signaturgesetz und der Signaturverordnung (Übersicht über geeignete Algorithmen)*, Stand: 2016-03-17, Veröffentlicht: BAnz AT 14.04.2016 B11, Bundesnetzagentur für Elektrizität, Gas, Telekommunikation, Post und Eisenbahnen.

[ANSSI 2011] *ANSSI 2011: http://www.legifrance.gouv.fr/affichTexte.do;jsessionid=?cidTexte=JORFTEXT000024668816*

[ANSI X9.62-1999] *ANSI X9.62-1999: Public Key Cryptography for the Financial Services Industry: the Elliptic Curve Digital Signature Algorithm (ECDSA)*, American National Standards Institute (ANSI), 1999.

[ANSI X9.62-2005] *ANSI X9.62-2005: Public Key Cryptography for the Financial Services Industry: the Elliptic Curve Digital Signature Algorithm (ECDSA)*, American National Standards Institute (ANSI), 2005.

[ANSI X9.63] *ANSI X9.63: Public Key Cryptography for the Financial Services Industry, Key Agreement and Key Transport Using Elliptic Curve cryptography*, American National Standards Institute (ANSI), January 2011.

[BN] *TCG Algorithm Registry/Family "2.0": Level 00 Revision 01.22*, February 9, 2015.

[RFC 5639] *RFC 5639: J. Merkle, ECC Brainpool Standard Curves and Curve Generation*, BSI, March 2010.

[SEC 2] *SEC 2: Standards for Efficient Cryptography, SEC 2: Recommended Elliptic Curve Domain Parameters*, Certicom Research, Version 1.0, September 20, 2000.

[TU] *TU Darmstadt: Cryptographically secure elliptic curves over GF(p) generated with complex multiplication by our Elliptic Curve Cryptogrphy Group with the OID prefix 1.3.6.1.4.1.8301.3.1.2.9.0*, http://www.flexiprovider.de/CurveOIDs.html

[FIPS 180-4] *FIPS PUB 180-4 Federal Information Processing Standards Publication Secure Hash Standard (SHS),* August 2015, Information Technology Laboratory National Institute of Standards and Technology.

[FIPS 186-4] *FIPS PUB 186-4 Federal Information Processing Standards Publication Digital Signature Standard (DSS)*, July 2013, Information Technology Laboratory National Institute of Standards and Technology.

[FIPS 197] *Federal Information Processing Standards Publication 197, Announcing the ADVANCED ENCRYPTION STANDARD (AES),* 2001-11-26, National Institute of Standards and Technology (NIST).

[IEEE Std 1363] *IEEE Std 1363™-2000: IEEE Standard Specifications for Public-Key Cryptography*, 2005-12-12, IEEE Computer Society.

[ISO/IEC 14888-3] *ISO/IEC 14888-3:2015: Information technology – Security techniques – Digital signatures with appendix - Part 3: Discrete logarithm based mechanisms*, 2016.

[ISO/IEC 9797-1] *ISO 9797-1: Information technology – Security techniques - Message Authentication Codes (MACs) - Part 1: Mechanisms using a block cipher*, 1999-12, ISO/IEC.

[ISO 11568-4] *ISO 11568-4: Banking – Key management (retail) – Part 4: Asymmetric cryptosystems – Key management and life cycle*, 2007

[ISO/IEC 11770-3] *ISO/IEC 11770-3:2015: Information technology – Security techniques - Key management - Part 3: Mechanisms using asymmetric techniques*, 2015, ISO/IEC.

[NIST SP 800-38A] *NIST Special Publication 800-38A*, *Recommendation for BlockCipher Modes of Operation* , National Institute of Standards and Technology (NIST), Technology Administration, U.S. Department of Commerce.

[NIST SP 800-38B] *NIST Special Publication 800-38B*, *Recommendation for BlockCipher Modes of Operation: The CMAC Mode for Authentication*, National Institute of Standards and Technology (NIST), Technology Administration, U.S. Department of Commerce.

[NIST SP 800-67] *NIST Special Publication 800-67 –Revision 2, Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher – Published November 2017*, National Institute of Standards and Technology (NIST), Technology Administration, U.S. Department of Commerce.

[PKCS #1] *PKCS #1: RSA Cryptography Standard*, Version 2.2, October 27, 2012, RSA Laboratories.

[TR-03110-1] *BSI TR-03110-1 Advanced Security Mechanisms for Machine Readable Travel Documents - Part 1: eMRTDs with BAC/PAVEv2 and EACv1*, Version 2.20, February 26, 2015, Bundesamt für Sicherheit in der Informationstechnik, Germany.

[TR-03110-2] *BSI TR-03110-2 Advanced Security Mechanisms for Machine Readable Travel Documents - Part 2: Protocols for electronic IDentification, Authentication and trust Services (eIDAS)*, Version 2.21, December 21, 2016, Bundesamt für Sicherheit in der Informationstechnik, Germany.

[TR-03110-3] *BSI TR-03110-3 Advanced Security Mechanisms for Machine Readable Travel Documents - Part 3: Common Specifications*, Version 2.21, December 21, 2016, Bundesamt für Sicherheit in der Informationstechnik, Germany.

[TR-03110-4] *BSI TR-03110-4 Advanced Security Mechanisms for Machine Readable Travel Documents - Part 4: Applications and Document Profiles*, Version 2.21, December 21, 2016, Bundesamt für Sicherheit in der Informationstechnik, Germany.

9 Legal information

9.1 Definitions

Draft — The document is a draft version only. The content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included herein and shall have no liability for the consequences of use of such information.

9.2 Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors. In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory. Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

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

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification. Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products. NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Limiting values — Stress above one or more limiting values (as defined in the Absolute Maximum Ratings System of IEC 60134) will cause permanent damage to the device. Limiting values are stress ratings only and (proper) operation of the device at these or any other conditions above those given in the Recommended operating conditions section (if present) or the Characteristics sections of this document is not warranted. Constant or repeated exposure to limiting values will permanently and irreversibly affect the quality and reliability of the device.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at http://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

No offer to sell or license — Nothing in this document may be interpreted or construed as an offer to sell products that is open for acceptance or the grant, conveyance or implication of any license under any copyrights, patents or other industrial or intellectual property rights.

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

Non-automotive qualified products — Unless this data sheet expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications. In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

Translations — A non-English (translated) version of a document is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

9.3 Trademarks

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

MIFARE — is a trademark of NXP B.V.

DESFire — is a trademark of NXP B.V.

Tables

Figures

Contents

Please be aware that important notices concerning this document and the product(s) described herein, have been included in section 'Legal information'.

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 31 May 2019