MF3Dx2

MIFARE DESFire EV2 – Security Target Lite

Rev. 1.5 – 2016-04-29 **Evaluation documentation**

Final **PUBLIC**

Document Information

Info	Content
Keywords	Security Target Lite, NXP Secure Smart Card Controller MF3Dx2 with IC Dedicated Support Software
Abstract	Evaluation of the MF3Dx2 developed and provided by NXP Semiconductors, Business Unit Identification, according to the Common Criteria for Information Technology Evaluation Version 3.1 at EAL5 augmented

Rev	Date	Description
1.0	29-June-2015	Initial Version of this Security Target Lite
1.1	06-October-2015	Added Threat Masquerade_TOE
1.2	10-December-2015	Update of Release and Date Information of the Components of the TOE
1.3	05-February-2016	Update of Document References
1.4	21-April-2016	Incorporated feedback from CB
1.5	29-April-2016	Update of Document References

1 ST Introduction

This chapter is divided into the following sections: ”ST Reference”, ”TOE Reference”, ”TOE Overview” and ”TOE Description”.

1.1 ST Reference

MF3Dx2 Security Target, 1.5, NXP Semiconductors, 2016-04-29.

1.2 TOE Reference

NXP Secure Smart Card Controller MF3Dx2, Version 1.5

1.3 TOE Overview

1.3.1 Introduction

NXP has developed the MF3Dx2 to be used with Proximity Coupling Devices (PCDs, also called "terminal") according to ISO14443 Type A [16]. The communication protocol complies to part ISO 14443-4 [15]. The MF3Dx2 is primarily designed for secure contact-less transport applications and related loyalty programs as well as access control systems as well as closed loop payment systems. It fully complies with the requirements for fast and highly secure data transmission, flexible memory organisation and interoperability with existing infrastructure.

The TOE is a Smart Card comprising a hardware platform and a fixed software package. The software package is stored in non-volatile memory and provides an operating system with a set of functions, used to manage the various kinds of data files stored in the non-volatile EEPROM memory. The operating system supports a separation between the data of different applications and provides access control if required by the configuration.

The TOE includes also IC Dedicated Software to support its start-up and for test purposes after production. The Smart Card Controller hardware comprises an 16-bit processing unit, volatile and non-volatile memories, cryptographic co-processors, security components and one communication interface.

The TOE includes a functional specification and a guidance document. This documentation contains a description of the hardware and software interface, the secure configuration and usage of the product by the terminal designer.

The security measures of the MF3Dx2 are designed to act as an integral part of the combination of hardware platform and software package in order to strengthen the product as a whole. Several security measures are completely implemented in and controlled by the hardware. Other security measures are controlled by the combination of hardware and software or software guided exceptions.

The different (package) types are described in detail in section 1.4.1.1.

1.3.2 TOE Type

The TOE is a Smart Card comprising a hardware platform and a fixed software package. The guidance consists of two documents that are also part of the TOE.

1.3.3 Required non-TOE Hardware/Software/Firmware

The TOE requires an ISO 14443 [14, 16, 17, 15] card terminal to be provided with power and to receive adequate commands.

1.4 TOE Description

1.4.1 Physical Scope of TOE

The Target of Evaluation (TOE) is the smartcard integrated circuit named MF3Dx2 in combination with a fixed software package, the IC Dedicated Software. The TOE is manufactured in an advanced CMOS process. The TOE includes IC Designer/Manufacturer proprietary IC Dedicated Test Software and IC Dedicated Support Software, according to the terminology used in [13]. Note that the MF3Dx2 Software is part of the IC Dedicated Support Software.

Table 1.1 list the TOE components.

Type	Name	Release	Date	Form of Delivery
Hardware	MF3Dx2 Hardware	VA,VB	11.06.2015	Wafer, modules and package
Software	Test ROM Software (the IC Dedicated Test Software)	1.0	25.06.2015	SM ROM on chip
Software	IC Dedicated Boot Software (part of the IC Dedicated Support Software)	1.0	25.06.2015	SM ROM on chip
Software	HAL ROM Software (part of the IC Dedicated Support Software)	1.0	25.06.2015	SM ROM on chip
Software	MIFARE DESFire Software (part of the IC Dedicated Support Software)	1.0	02.11.2015	SM ROM on chip
Document	MF3Dx2 - MIFARE DESFire EV2 contactless multi-application IC, Product Data Sheet, [10]	226030	2016-02-04	Electronic Document
Document	MF3Dx2 - Information on Guidance and Operation, Guidance and Operation Manual [8]	274811	2016-04-29	Electronic Document

Tab. 1.1: Components of the TOE

1.4.1.1 Evaluated Chip and Package Types

A number of package types are supported for the TOE. Each package type has a different commercial type name. The TOE will be available in two package types and four different memory configurations.
A commercial type name for the TOE has the following general format:

- MF3D*cxeffdpp*/*fv*

Type	*c*	*x*	*e*	*ff*	*d*	*pp*	/	*f*	*v*
MF3D		4	2	01	D	UD	/	0	0
...	...	...	...	...	...	...	/	...	...

Tab. 1.2: Supported Types

Table 1.2 illustrates the commercial type names that are subject of the evaluation.

Identifier	Description	Valid Values	Digits	Assignment	Meaning
c	input capacitance	alpahabetic	1 – 2	”	17 pF
				H	70 pF
x	memory size	numeric	1	0	0.5KB EEPROM
				2	2KB EEPROM
				4	4KB EEPROM
				8	8KB EEPROM
e	evolution	numeric	1	2	the third evolution of MIFARE DESFire
ff	FAB produced	numeric	2	00	Multiple Fabs
				01	SSMC
d	operating temperature range	alphabetic	1	D	$-20 < t_{\text{operating}} < 70$
pp	package type	alphabetic	2	UD	$120\mu m$ sawn wafer
				UF	$75\mu m$ sawn wafer
				A4	MOA4 module on reel
				A6	MOB6 module on reel
f	Fabkey Identifier	alphanumeric	1	0	Default EEPROM configuration
				1..9,A..Z	Dedicated customer specific EEPROM configuration
v	Product Revision	alphanumeric	1	0	Revision 1

Tab. 1.3: Variable Definitions for Commercial Type Names

The package type does not influence the security functionality of the TOE. For all package types listed above the security during development and production is ensured (refer to section 1.4.3).
All commercial types listed in the table above are subject of this evaluation. However the identifier ”MF3Dx2” will be used in the remainder of the document to make referencing easier. Unless described explicitly all information

given in the remainder of the ST applies to all commercial types.

1.4.2 Logical Scope of TOE

1.4.2.1 Hardware Description

The CPU of the MF3Dx2 has an 16-bit architecture. The on-chip hardware components are controlled by the MIFARE DESFire Software via Special Function Registers. These registers are correlated to the activities of the CPU, the memory management unit, interrupt control, contact-less communication, EEPROM, timers, the DES co-processor and the AES co-processor. The communication with the MF3Dx2 can be performed through the contact-less interface.
The device includes ROM (48 kByte), RAM (1 kByte), EEPROM (10 kByte) and FLASH (64 kByte) memory. The ROM is split in Application-ROM, HAL-ROM and Test-ROM. The EEPROM size can be logically configured as denoted in Table 1.3.
The unified AES/Triple-DES co-processor supports AES operations with a key length of 128 bits and Triple-DES operations with key lengths of 112 bits and 168 bits. The random number generator provides true random numbers which are used to seed pseudo random number generator.

1.4.2.2 Software Description

The IC Dedicated Test Software (Test ROM Software) in the Test-ROM of the TOE is used by the TOE Manufacturer to test the functionality of the chip. The test functionality is disabled before the operational use of the smart card. The IC Dedicated Test Software includes the test operating system, test routines for the various blocks of the circuitry, control flags for the status of the EEPROM security row and shutdown functions to ensure that security relevant test operations cannot be executed illegally after phase 3 of the TOE Life cycle (see Section 1.4.4).

The TOE also contains IC Dedicated Support Software. The Boot ROM Software which is stored in the Test-ROM is part of the IC Dedicated Support Software. This 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.
The MIFARE DESFire Software is also part of the IC Dedicated Support Software and provides the main functionality of the TOE in the usage phase. The MF3Dx2 is primarily designed for secure contact-less transport applications and related loyalty programs as well as access control systems. It fully complies with the requirements for fast and highly secure data transmission, flexible memory organization and interoperability with existing infrastructure. Its functionality consists of:

- Flexible file system that groups user data into applications and files within each application.
- Support for different file types like values or data records.
- Mutual three pass authentication, also according to ISO 7816-4.
- Authentication on application level with fine-grained access conditions for files.

- Multi-application support that allows distributed management of applications and ensures application segregation.
- Delegated-application support that allows third party service providers to create their applications onto the issued TOE.
- Multiple application selection that allows transaction over files in two applications.
- Data encryption on the communication path.
- Message Authentication Codes (MAC) for replay attack protection.
- Transaction system with rollback that ensures consistency for complex transactions.
- Unique serial number for each device (UID) with optional random UID.
- Key set rolling feature per application to switch to a predefined key set.
- Transaction MAC feature to prevent fraudulent merchant attacks.
- Originality functionality that allows verifying the authenticity of the TOE.
- Virtual Card architecture to allow multiple applications on one device.
- Proximity check feature against relay attacks on the TOE.
- The TOE supports a MIFARE DESFire D40 backward compatible mode for authentication. The backward compatible mode for authentication is not part of any Security Functional Requirement of this Security Target and is therefore not in the scope of the evaluation.
- The TOE supports a MIFARE DESFire EV1 backward compatible authentication with 2-key Triple-DES. 2-key Triple-DES authentication is not part of any Security Functional Requirement of this Security Target and is therefore not in the scope of the evaluation.

The TOE features enable it to be used for a variety of applications:

- Electronic fare collection
- Stored value card systems
- Access control systems
- Loyalty

If privacy is an issue, the TOE can be configured not to disclose any information to unauthorized users.

1.4.2.3 Documentation

The Functional Specification [10] is also part of the TOE. It contains a functional description of the communication protocol and the commands implemented by the TOE. The provided documentation can be used by a customer to construct applications using the TOE.

The Functional Specification is supported by the Application Note "MF3Dx2 - Information on Guidance and Operation" [8] which gives additional guidance with regard to the secure usage of the TOE.

1.4.3 Security during Development and Production

During the design and the layout process of the IC and the development of the software only people involved in the specific development project have access to sensitive data. The security measures installed within NXP ensure a secure computer system and provide appropriate equipment for the different development tasks.
The verified layout data is provided by the developers of NXP Semiconductors, Business Unit Identification directly to the wafer fab. The wafer fab generates and forwards the layout data related to the different photo masks to the manufacturer of the photo masks. The photo masks are generated off-site and verified against the design data of the development before the usage. The accountability and the traceability is ensured among the wafer fab and the photo mask provider.
The test process of every die is performed by a test center of NXP. Delivery processes between the involved sites provide accountability and traceability of the produced wafers. NXP embeds the dice into smartcard modules based on customer demand. Information about non-functional items is stored on magnetic/optical media enclosed with the delivery, available for download or the non-functional items are physically marked.
In summary the TOE can be delivered in two different forms:

- Dice on wafers
- Smart Card Modules on a module reel

The different (package) types are described in detail in section 1.4.1.1

1.4.4 Life Cycle and Delivery of the TOE

The life-cycle phases are according to the Security IC Platform Protection Profile with Augmentation Packages [13], section 1.2.4:

- Phase 1: IC Embedded Software Development
- Phase 2: IC Development
- Phase 3: IC Manufacturing
- Phase 4: IC Packaging
- Phase 5: Composite Product Integration
- Phase 6: Personalisation
- Phase 7: Operational Usage

For the usage phase the MF3Dx2 chip will be embedded in a credit card (meaning ID-1 sized) plastic card (micro-module embedded into the plastic card) or another sealed package. The module and card embedding of the TOE provide external security mechanisms because they make it harder for an attacker to access parts of the

TOE for physical manipulation.

Regarding the Application Note 1 of [13], NXP will deliver the TOE at the end of Phase 3 in form of wafers or at the end of Phase 4 in packaged form. Therefor the TOE evaluation perimeter comprising the development and production environment of the TOE, consists of life-cycle phases 2 - 4 (according to the Security IC Platform Protection Profile with Augmentation Packages [13], section 1.2.4).

Regarding the Application Note 2 of [13] the TOE provides additional functionality which is not covered in the Security IC Platform Protection Profile with Augmentation Packages [13]. The additional functionality is due to the MIFARE DESFire Software that is part of the IC Dedicated Support Software and included in this evaluation. The MIFARE DESFire Software is embedded in the TOE during the TOE evaluation perimeter (life-cycle phases 2 - 4) and the TOE does not allow the application of any IC Embedded Software after TOE delivery. Moreover, the TOE is getting locked before TOE delivery at the end of Phase 3 or Phase 4.

The TOE is able to control two different logical phases. After production of the chip every start-up will lead to the Test Mode and the execution of the IC Dedicated Test Software. At the end of the production test the access to the IC Dedicated Test Software is disabled. With disabled test software every start-up of the chip will lead to the User Mode with the CPU executing the MIFARE DESFire Software.

1.4.5 TOE Intended Usage

The TOE user environment is the environment from TOE Delivery to Phase 7. At the phases up to 6, the TOE user environment must be a controlled environment. Regarding to Phase 7, the TOE is used by the end-user. The method of use of the product in this phase depends on the application. The TOE is intended to be used in an unsecured environment that does not avoid a threat.

The device is developed for high-end safeguarded applications, and is designed for embedding into contact-less smart cards according to ISO 14443 [14, 16, 17, 15]. Usually the smart card is assigned to a single individual only and the smart card may be used for multiple applications in a multi-provider environment. The secret data shall be used as input for the calculation of authentication data, encryption and integrity protection of data for communication.

In the end-user environment (Phase 7) Smart card ICs are used in a wide range of applications to assure authorized conditional access. Examples of such are transportation or access management. The end-user environment therefore covers a wide spectrum of very different functions, thus making it difficult to avoid and monitor any abuse of the TOE.

The system integrators such as the terminal software developer may use samples of the TOE during the development phases for their testing purposes. These samples do not differ from the TOE, they do not have any additional functionality used for testing.

Remark 1. The phases from TOE Delivery to Phase 7 of the smart card life cycle are not part of the TOE construction process in the sense of this Security Target. Information about those phases is just included to describe how the TOE is used after its construction. Nevertheless the security features of the TOE cannot be disabled in these phases.

1.4.6 Interface of the TOE

The electrical interface of the TOE are the pads to connect the RF antenna. The functional interface is defined by the commands implemented by the TOE and described in [10].

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 where the attacker e.g. manipulates the chip surface.

2 Conformance Claims

2.1 CC Conformance Claim

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

- Common Criteria for Information Technology Security Evaluation, Part 1 – Introduction and general model - Version 3.1 CCMB-2012-09-001, Revision 4, September 2012, [2]
- Common Criteria for Information Technology Security Evaluation, Part 2 – Security functional components, Version 3.1 CCMB-2012-09-002, Revision 4, September 2012, [3]
- Common Criteria for Information Technology Security Evaluation, Part 3 – Security Assurance Components, Version 3.1 CCMB-2012-09-003, Revision 4, September 2012, [4]

For the evaluation the following methodology will be used:

- Common Methodology for Information Technology Security Evaluation – Evaluation Methodology, Version 3.1 CCMB-2012-09-004, Revision 4, September 2012, [5]

This Security Target claims to be CC Part 2 extended and CC Part 3 conformant. The extended Security Functional Requirements are defined in chapter 6.

2.2 Package Claim

This Security Target claims conformance to the assurance package **EAL5 augmented**. The augmentations to EAL5 are ALC_DVS.2 and AVA_VAN.5. In addition, this Security Target is augmented using the components ASE_TSS.2 and ALC_FLR.1.

Note: The Protection Profile Security IC Platform Protection Profile with Augmentation Packages [13], to which this Security Target claims conformance (refer to section 2.3), requires assurance level EAL4 augmented. The changes, which are needed for EAL5, are described in the relevant sections of this Security Target.

2.3 PP Claim

This Security Target claims strict conformance to the Security IC Platform Protection Profile with Augmentation Packages [13]. Thus, the concepts are used in the same sense. For the definition of terms refer to [13]. This chapter does not need any supplement in the Security Target.

Note that the Security IC Platform Protection Profile with Augmentation Packages [13] defines (optional) ”Augmentation Packages”, which are not applied in this Security Target.

2.4 Conformance Claim Rationale

According to section 2.3 this Security Target claims strict conformance to the Security IC Platform Protection Profile with Augmentation Packages [13]. Note that the term Protection Profile will be used in the remainder of the document to make referencing easier.

The TOE type defined in section 1.3.2 of this Security Target is a smart card controller with IC Dedicated Support Software. This is consistent with the TOE definition for a Security IC in section 1.2.2 of [13].

The sections within this document where security problem definitions, objectives and security requirements are defined, clearly state which of these items are taken from the Protection Profile and which are added in this ST. Therefore the content of the Protection Profile is not repeated in this Security Target. Moreover, all additionally stated items in this Security Target do not contradict the items included from the Protection Profile (see the respective sections in this document). The operations done for the SFRs taken from the Protection Profile are also clearly indicated.

The evaluation assurance level claimed for this TOE (EAL5 augmented) is shown in section 6.2 to include respectively exceed the requirements claimed by the Protection Profile (EAL4 augmented).

These considerations show that the Security Target correctly claims conformance to the Protection Profile.

3 Security Problem Definition

Since this Security Target claims strict conformance to the Protection Profile, The Assets, Threats, Assumptions, and Organizational Security Policies are taken from the Protection Profile. In the following only the extensions of the different sections are detailed. The elements of the Security Problem Definition that are not extended in the Security Target are not repeated in this Security Target, they are cited here for completeness only.

3.1 Description of Assets

All assets, which are related to the high-level concerns defined in section 3.1 of the Protection Profile, are related to standard functionality and are applied in this Security Target. The high-level concerns are cited here completely:

- Integrity and confidentiality of User Data stored and in operation,
- Integrity and confidentiality of the Security IC Embedded Software, stored and in operation,
- Correct operation of the Security Services provided by the TOE for the Security IC Embedded Software,
- Deficiency of random numbers.

To be able to protect the assets based on this concerns, the TOE shall protect its security functionality. Therefore, critical information about the TOE shall be protected. Critical information includes:

- Logical design data, physical design data, IC Dedicated Software, Security IC Embedded Software and configuration data.
- Initialization Data and Pre-personalization Data, specific development aids, test and characterization related data, material for software development support, and photo masks.

Note that the keys for the cryptographic co-processors are seen as User Data.

3.2 Threats

All threats, defined in section 3.2 of the Protection Profile, are valid for this Security Target. These threats are listed in table 3.1. In addition the threat T.Masquerade_TOE is applicable for this TOE as stated below.

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

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
T.Masquerade_TOE	Masquerade the TOE

Tab. 3.1: Threats defined in the Security IC Protection Profile

Considering the Application Note 4 in the Protection Profile, the following additional threats are defined in this Security Target:

Name	Title
T.Data-Modification	Unauthorised Data Modification
T.Impersonate	Impersonating authorised users during authentication
T.Cloning	Cloning

Tab. 3.2: Additional Threats defined in this ST

T.Data-Modification **Unauthorised Data Modification**
User data stored by the TOE may be modified by unauthorised subjects. This threat applies to the processing of modification commands received by the TOE, it is not concerned with verification of authenticity.

T.Impersonate **Impersonating authorised users during authentication**
An unauthorised subject may try to impersonate an authorised subject during the authentication sequence, e.g. by a man-in-the middle or replay attack.

T.Cloning **Cloning**
User and TSF data stored on the TOE (including keys) may be read out by an unauthorised subject in order to create a duplicate.

3.3 Organizational Security Policies

All security policies defined in section 3.3 of the Protection Profile are valid for this Security Target. These security policies are listed in Table 3.3.

Name	Title
P.Process-TOE	Identification during TOE Development and Production

Tab. 3.3: Policies defined in the Security IC Protection Profile

In compliance with Application Note 5 in the Protection Profile, this Security Target defines additional security policies as detailed in the following.

The TOE provides specific security functionality which can be used by the MIFARE DESFire Software. In the following specific security functionality is listed which is not derived from threats identified for the TOE's environment because it can only be decided in the context of the smart card application, against which threats the MIFARE DESFire Software will use the specific security functionality.

The IC Developer / Manufacturer therefore applies the policies Confidentiality during communication, Integrity during communication, Transaction mechanism and Un-traceability of end-users as specified below.

Name	Title
P.Encryption	Confidentiality during communication
P.MAC	Integrity during communication
P.Transaction	Transaction mechanism
P.No-Trace	Un-traceability of end-users

Tab. 3.4: Additional Policies defined in this ST

P.Encryption **Confidentiality during communication**
The TOE shall provide the possibility to protect selected data elements from eavesdropping during contact-less communication.

P.MAC **Integrity during communication**
The TOE shall provide the possibility to protect the contact-less communication from modification or injections. This includes especially the possibility to detect replay or man-in-the-middle attacks within a session.

P.Transaction **Transaction mechanism**
The TOE shall provide the possibility to combine a number of data modification operations in one transaction, so that either all operations or no operation at all is performed.

P.No-Trace **Un-traceability of end-users**
The TOE shall provide the ability that authorised subjects can prevent that end-user of TOE may be traced by unauthorised subjects without consent. Tracing of end-users may happen by performing a contact-less communication with the TOE when the end-user is not aware of it. Typically this involves retrieving the UID or any freely accessible data element.

3.4 Assumptions

All assumptions defined in section 3.4 of the Protection Profile are valid for this Security Target. These assumptions are listed in Table 3.5.

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

Tab. 3.5: Assumptions defined in the Security IC Protection Profile

In compliance with Application Notes 6 and 7 in the Protection Profile, this Security Target defines two additional assumptions as follows.

A.Secure_Values **Usage of secure values**
Only confidential and secure cryptographically strong keys shall be used to set up the authentication. These values are generated outside the TOE and they are downloaded to the TOE.

A.Terminal_Support **Terminal support to ensure integrity, confidentiality and use of random numbers**
The terminal verifies information sent by the TOE in order to ensure integrity and confidentiality of the communication. Furthermore the terminal shall provide random numbers according to AIS20 (see [18]) or AIS31 (see [19]) for the authentication.

These assumptions are summarized in Table 3.6.

Name	Title
A.Secure_Values	Usage of secure values
A.Terminal_Support	Terminal support to ensure integrity, confidentiality and use of random numbers

Tab. 3.6: Additional Assumptions defined in this ST

4 Security Objectives

4.1 Security Objectives for the TOE

All security objectives for the TOE, which are defined in section 4.1 of the Protection Profile, are applied to this Security Target and listed in table 4.1.

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

Tab. 4.1: Security Objectives of the TOE (PP)

Regarding the Application Notes 8 and 9 in the Protection Profile, additional security objectives that are based on additional functionality provided by the TOE, are defined and listed in table 4.2.

Name	Title
O.Access-Control	Access Control
O.Authentication	Authentication
O.Encryption	Confidential Communication
O.MAC	Integrity-protected Communication
O.Type_Consistency	Data type consistency
O.Transaction	Transaction mechanism
O.No-Trace	Preventing Traceability

Tab. 4.2: Security Objectives of the TOE (ST)

These additional security objectives are specified as follows.

O.Access-Control **Access Control**

The TOE must provide an access control mechanism for data stored by it. The access control mechanism shall apply to read, modify, create and delete operations for data elements and to reading and modifying security attributes as well as authentication data. It shall be possible to

limit the right to perform a specific operation to a specific user. The security attributes (keys) used for authentication shall never be output.

O.Authentication **Authentication**
The TOE must provide an authentication mechanism in order to be able to authenticate authorised users. The authentication mechanism shall be resistant against replay and man-in-the-middle attacks.

O.Encryption **Confidential Communication**
The TOE must be able to protect the communication by encryption. This shall be implemented by security attributes that enforce encrypted communication for the respective data elements.

O.MAC **Integrity-protected Communication**
The TOE must be able to protect the communication by adding a MAC. This shall be implemented by security attributes that enforce integrity protected communication for the respective data elements. Usage of the protected communication shall also support the detection of injected and bogus commands within the communication session before the protected data transfer.

O.Type_Consistency **Data type consistency**
The TOE must provide a consistent handling of the different supported data types. This comprises over- and underflow checking for values, for data file sizes and record handling.

O.Transaction **Transaction mechanism**
The TOE must be able to provide a transaction mechanism that allows to update multiple data elements either all in common or none of them.

O.No-Trace **Preventing Traceability**
The TOE must be able to prevent that the TOE end-user can be traced. This shall be done by providing an option that disables the transfer of any information that is suitable for tracing an end-user by an unauthorised subject.

4.2 Security Objectives for the Security IC Embedded Software development Environment

All security objectives for the Security IC Embedded Software development Environment, which are defined in section 4.2 of the Protection Profile, are applied to this Security Target and listed in table 4.3.

Name	Title
OE.Resp-Appl	Treatment of User Data

Tab. 4.3: Security Objectives of the DVE (PP)

Clarification related to "Treatment of User Data (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.

4.3 Security Objectives for the Operational Environment

In addition to the security objective for the operational environment as required by CC Part 1 [2], all security objectives for the operational environment, which are defined in section 4.3 of the Protection Profile, are applied to this Security Target and listed in table 4.4.

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

Tab. 4.4: Security Objectives of the OPE (PP)

In addition, the following additional security objectives for the operational environment are defined in this Security Target and listed in table 4.5.

Name	Title
OE.Secure_Values	Generation of secure values
OE.Terminal_Support	Terminal support to ensure integrity, confidentiality and use of random numbers

Tab. 4.5: Security Objectives of the OPE (ST)

The TOE provides specific functionality that requires the TOE Manufacturer to implement measures for the unique identification of the TOE. Therefore, OE.Secure_Values is defined to allow a TOE specific implementation (refer also to A.Secure_Values).

OE.Secure_Values **Generation of secure values**
The environment shall generate confidential and cryptographically strong keys for authentication purpose. These values are generated outside the TOE and they are downloaded to the TOE during the personalisation or usage in phase 5 to 7

The TOE provides specific functionality to verify the success of the application download process. Therefore,

OE.Terminal_Support is defined to allow triggering the verification process.

OE.Terminal_Support **Terminal support to ensure integrity, confidentiality and use of random numbers**
The terminal shall verify information sent by the TOE in order to ensure integrity and confidentiality of the communication. This involves checking of MAC values, verification of redundancy information according to the cryptographic protocol and secure closing of the communication session. Furthermore the terminal shall provide random numbers according to AIS20 (see [18]) or AIS31 (see [19]) for the authentication.

4.4 Security Objectives Rationale

Section 4.4 in the Protection Profile provides a rationale how the threats, organisational security policies and assumptions are addressed by the security objectives defined in the Protection Profile. Table 4.6 summarizes this.

Security Problem Definition	Security Objective	Notes
T.Leak-Inherent	O.Leak-Inherent	
T.Phys-Probing	O.Phys-Probing	
T.Malfunction	O.Malfunction	
T.Phys-Manipulation	O.Phys-Manipulation	
T.Leak-Forced	O.Leak-Forced	
T.Abuse-Func	O.Abuse-Func	
T.RND	O.RND	
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
T.Masquerade_TOE	OE.Process-Sec-IC	

Tab. 4.6: Security Objectives vs. Security Problem Definition (PP)

Table 4.7 summarizes how threats, organisational security policies and assumptions are addressed by the security objectives with respect to those items defined in the Security Target. All these items are in line with those in the Protection Profile.

Security Problem Definition	Security Objective	Notes
T.Data-Modification	O.Access-Control O.Type_Consistency OE.Terminal_Support	
T.Impersonate	O.Authentication	
T.Cloning	O.Access-Control O.Authentication	

Security Problem Definition	Security Objective	Notes
P.Encryption	O.Encryption	
P.MAC	O.MAC	
P.Transaction	O.Transaction	
P.No-Trace	O.Access-Control O.Authentication O.No-Trace	
A.Secure_Values	OE.Secure_Values	
A.Terminal_Support	OE.Terminal_Support	

Tab. 4.7: Security Objectives vs. Security Problem Definition (ST)

The rationale for the threat T.Masquerade_TOE is given below:

Justification related to T.Masquerade_TOE:

Objective	Rationale
OE.Process-Sec-IC	The Security Objective for the Operational Environment requires that the confidentiality and integrity of the TOE is maintained. Thus the threat is covered.

The rationale for all items defined in the Security Target is given below.

Justification related to T.Data-Modification:

Objective	Rationale
O.Access-Control	This objective requires an access control mechanism that limits the ability to modify data and code elements stored by the TOE.
O.Type_Consistency	This objective ensures that data types are adhered, so that TOE data can not be modified by abusing type-specific operations.
OE.Terminal_Support	This objective requires that the terminal must support this by checking the TOE responses.

Justification related to T.Impersonate:

Objective	Rationale
O.Authentication	This objective requires that the authentication mechanism provided by the TOE shall be resistant against attack scenarios targeting the impersonation of authorized users.

Justification related to T.Cloning:

Objective	Rationale
O.Access-Control	This objective requires that unauthorized users can not read any information that is restricted to the authorized subjects. The cryptographic keys used for the authentication are stored inside the TOE and are protected by this objective. This objective states that no keys used for authentication shall ever be output.
O.Authentication	This objective requires that users are authenticated before they can read any information that is restricted to authorized users.

Justification related to A.Secure_Values:

Objective	Rationale
OE.Secure_Values	This objective is an immediate transformation of the assumption, therefore it covers the assumption.

Justification related to A.Terminal_Support:

Objective	Rationale
OE.Terminal_Support	This objective is an immediate transformation of the assumption, therefore it covers the assumption. The TOE can only check the integrity of data received from the terminal. For data transferred to the terminal the receiver must verify the integrity of the received data. Furthermore the TOE cannot verify the entropy of the random number sent by the terminal. The terminal itself must ensure that random numbers are generated with appropriate entropy for the authentication. This is assumed by the related assumption, therefore the assumption is covered.

Justification related to P.Encryption:

Objective	Rationale
O.Encryption	This objective is an immediate transformation of the security policy, therefore it covers the Security policy.

Justification related to P.MAC:

Objective	Rationale
O.MAC	This objective is an immediate transformation of the security policy, therefore it covers the Security policy.

Justification related to P.Transaction:

Objective	Rationale
O.Transaction	This objective is an immediate transformation of the security policy, therefore it covers the Security policy.

Justification related to P.No-Trace:

Objective	Rationale
O.Access-Control	This objective provides means to implement access control to data elements on the TOE in order to prevent tracing based on freely accessible data elements.
O.Authentication	This objective provides means to implement authentication on the TOE in order to prevent tracing based on freely accessible data elements.
O.No-Trace	This objective requires that the TOE shall provide an option to prevent the transfer of any information that is suitable for tracing an end-user by an unauthorized subject. This objective includes the UID.

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 Definitions

This Security Target does not define extended components.

Note that the Protection Profile defines extended security functional requirements FCS_RNG.1, FMT_LIM.1, FMT_LIM.2, FAU_SAS.1 and FDP_SDC.1 in chapter 5, which are included in this Security Target.

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 and the security assurance requirements 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 Part 1 [2]. These operations are used in the Protection Profile and in this Security Target, respectively.

The refinement operation is used to add details to requirements, and thus, further intensifies a requirement.

Refinements of security requirements are denoted in such a way that added words are in bold text.

The selection operation is used to select one or more options provided by the Protection Profile 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 italic text.

The iteration operation is used when a component is repeated with varying operations. It is denoted by showing brackets "'[iteration indicator]"' and the iteration indicator within the brackets.

For the sake of a better readability, the iteration operation may also be applied to some single components (being not repeated) in order to indicate belonging of such SFRs to same functional cluster. In such a case, the iteration operation is applied to only one single component.

Whenever an element in the Protection Profile contains an operation that is left uncompleted, the Security Target has to complete that operation.

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 Protection Profile and this Security Target.

6.1.1 SFRs of the Protection Profile

Table 6.1 shows all SFRs which are specified in the Protection Profile.

Name	Title
FAU_SAS.1[HW]	Audit Storage
FCS_RNG.1[HW]	Random Number Generation (Class PTG.2)
FCS_RNG.1[DET]	Random Number Generation (Deterministic)
FDP_ITT.1[HW]	Basic Internal Transfer Protection
FDP_IFC.1	Subset Information Flow Control
FDP_SDC.1[HW]	Stored data confidentiality
FDP_SDI.2[HW]	Stored data integrity monitoring and action
FMT_LIM.1[HW]	Limited Capabilities

Name	Title
FMT_LIM.2[HW]	Limited Availability
FPT_FLS.1	Failure with Preservation of Secure State
FPT_ITT.1[HW]	Basic Internal TSF Data Transfer Protection
FPT_PHP.3	Resistance to Physical Attack
FRU_FLT.2	Limited Fault Tolerance

Tab. 6.1: Security Functional Requirements defined in the Security IC Protection Profile

All assignment and selection operations of the SFR listed in the table above are performed except the operations completed below:

For the SFR FAU_SAS.1[HW] the Protection Profile leaves the assignment operation open for the non volatile memory type in which initialization data, pre-personalization data and/or other supplements for the Security IC Embedded Software are stored. This assignment operation is filled in by the following statement. Note that the assignment operations for the list of subjects and the list of audit information have already been filled in by the Protection Profile.

FAU_SAS.1[HW]	**Audit Storage**
Hierarchical-To	No other components.
Dependencies	No dependencies.
FAU_SAS.1.1[HW]	The TSF shall provide *the test process before TOE Delivery* with the capability to store *the Initialisation Data and/or Pre-personalisation Data and/or supplements of the Security IC Embedded Software* in the *NVM*.

For FCS_RNG.1.1 the Protection Profile partially fills in the assignment for the security capabilities of the RNG by requiring a total failure test of the random source and adds an assignment operation for additional security capabilities of the RNG. In addition, for FCS_RNG.1.2 the Protection Profile partially fills in the assignment operation for the defined quality metric for the random numbers by replacing it by a selection and assignment operation.

For the above operations the original operations defined in chapter 5 of the Protection Profile have been replaced by the open operations in the statement of the security requirements in chapter 6 of the Protection Profile for better readability. Note that the selection operation for the RNG type has already been filled in by the Protection Profile.

FCS_RNG.1[HW]	**Random Number Generation (Class PTG.2)**
Hierarchical-To	No other components.

Dependencies	No dependencies.
FCS_RNG.1.1[HW]	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[HW]	The TSF shall provide *octets of bits* that meet:

(PTG.2.6) Test procedure A [1] does not distinguish the internal random numbers from output sequences of an ideal RNG.

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

Note: The definition of the Security Functional Requirement FCS_RNG.1 has been taken from [1].

Note: The functional requirement FCS_RNG.1[HW] is a refinement of FCS_RNG.1 defined in PP [13] according to [1].

Note: Application Note 20 in [13] requires that the Security Target specifies for the security capabilities in FCS_RNG.1.1[HW] how the results of the total failure test of the random source are provided to the MIFARE DESFire Software. The results of the internal test sequence are provided to the MIFARE DESFire Software as a pass or fail criterion. The entropy of the random number is measured by the Shannon-Entropy as follows: $E = -\sum_{i=0}^{255} p_i \cdot \log_2 p_i$ where p_i is the probability that the byte $(b_7, b_6, \ldots, b_0)$ is equal to i as binary number. Here the term "bit" means measure of the Shannon-Entropy. The value "7.976" is assigned due to the requirements of "AIS31", [19].

In addition to FCS_RNG.1[HW] the TOE provides a deterministic random number generator:

FCS_RNG.1[DET]	**Random Number Generation (Deterministic)**
Hierarchical-To	No other components.

[1] Note: according par.295 in [19] the assignment may be empty.

Dependencies	No dependencies.
FCS_RNG.1.1[DET]	The TSF shall provide a *deterministic* random number generator that implements:
	(DRG.3.1) *If initialized with a random seed using a PTRNG of class PTG.2 (as defined in [19]) as random source, the internal state of the RNG shall have at least 230 bits (TDES) resp. 254 bits (AES) of entropy.*
	(DRG.3.2) *The RNG provides forward secrecy (as defined in [19]).*
	(DRG.3.3) *The RNG provides backward secrecy even if the current internal state is known (as defined in [19]).*
FCS_RNG.1.2[DET]	The TSF shall provide random numbers that meet:
	(DRG.3.4) *The RNG, initialized with a random seed using a PTRNG of class PTG.2 (as defined in [19]) as random source, generates output for which in AES mode* 2^{48} *and in 3DES mode* 2^{35} *strings of bit length 128 are mutually different with probability at least* $1 - 2^{-24}$ *in AES mode and* $1 - 2^{-17}$ *in 3DES mode.*
	(DRG.3.5) *Statistical test suites cannot practically distinguish the random numbers from output sequences of an ideal RNG. The random numbers must pass test procedure A* [2] *(as defined in [19]).*

Note: The CryptoLib Software provides the Security IC Embedded Software with seperate functionality to initialise the deterministic random number generator (which includes the chi-square test) and to generate pseudo-random data. It is the responsibility of the user to initialise the DRNG before generating random data. If it is tried to request pseudo-random numbers without having seeded the DRNG a security reset is triggered.

Note: Only if the chi-square test succeeds the hardware random number generator seeds the deterministic random number generator implemented as part of the CryptoLib Software.

For FDP_SDC.1.1 the Protection Profile leaves the assignment operation open for the memory area in which the TSF ensures the confidentiality of information of user data while being stored in that memory area. The assignment operation is filled with the following statement.

FDP_SDC.1[HW]	**Stored data confidentiality**
Hierarchical-To	No other components.
Dependencies	No dependencies.
FDP_SDC.1.1[HW]	The TSF shall ensure the confidentiality of the information of the user data while it is stored in the *RAM and Non Volatile Memory*.

For FDP_SDI.2.1 the Protection Profile leaves the assignment operations open on the type of integrity errors of user data and the attributes the user data is based on. For FDP_SDI.2.2 the Protection Profile leaves the

[2] Note: according par.295 in [19] the assignment may be empty.

assignment operation open on the type of action that shall be taken upon registration of integrity errors. The assignment operations are filled with the following statements.

FDP_SDI.2[HW]	**Stored data integrity monitoring and action**
Hierarchical-To	FDP_SDI.1 Stored data integrity monitoring
Dependencies	No dependencies.
FDP_SDI.2.1[HW]	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[HW]	Upon detection of a data integrity error, the TSF shall *trigger a Security Reset*.

By this, all assignment/selection operations are performed. This Security Target does not perform any other/-further operations for the Security Functional Requirements defined in the Protection Profile. Considering the Application Note 12 in the Protection Profile, in the following subsection the additional functions, such as for cryptographic support, authentication and access control are defined. These SFRs are not required in the Protection Profile. As required by the Application Note 14 in the Protection Profile, the secure state is described in section 7.2.1 in [9]. Regarding the Application Note 15 in the Protection Profile, an additional generation of audit is not defined for ”Limited Fault Tolerance (FRU_FLT.2)”. As required by the Application Note 19 in the Protection Profile, the automatic response of the TOE is described in section 7.2.1 in [9].

6.1.2 Additional SFRs regarding Access Control

6.1.2.1 Access Control Policy

The Security Function Policy (SFP) **Access Control Policy** uses the following definitions: The subjects are

Subject	**Admin**	**Administrator**
Info	The Admin is the subject that owns or has access to the PICCMasterKey.	
Info	The Admin is the subject that distributes the PICCDAMAuthKey, DAMMACs, and DAMENCs containing theAppDAMDefaultKey, to the DelAppMgr.	

Subject	**AppMgr**	**Application Manager**
Info	The AppMgr is the subject that owns or has access to an AppMasterKey. Note that the TOE supports multiple Applications and therefore multiple AppMgr, however for one Application there is only one AppMgr.	

Subject	DelAppMgr	Delegated Application Manager
Info	The DelAppMgr is the subject that has access to a valid DAMMAC, the PICCDAMAuthKey, and a DAMENC containing the AppDAMDefaultKey. Note that the TOE supports multipleDelApplications and therefore multiple DelAppMgr.	

Subject	AppUser	Application User
Info	The AppUser is the subject that owns or has access to an AppKey. Note that the TOE supports multiple AppUser within each Application and the assigned rights to the AppUser can be different, which allows to have more or less powerful AppUser.	

Subject	AppChangeUser	Application Change User
Info	The AppChangeUser is the subject that owns or has access to an AppChangeKey.	

Subject	AppRollUser	Application Roll Key Set User
Info	The AppRollUser is the subject that owns or has access to an AppRollKey.	

Subject	OrigKeyUser	Originality Key User
Info	The OrigKeyUser is the subject that owns or has acces to an OriginalityKey. The OrigKeyUser can authenticate with the TOE to prove the authenticity of the Security IC.	

Subject	Anybody	Anybody
Info	Any subject that does not belong to one of the roles Admin, AppMgr, DelAppMgr, AppUser, AppChangeUser, AppRollUser or OrigKeyUser, belongs to the role Anybody. This role includes the card holder (also referred to as end-user), and any other subject like an attacker for instance. The subjects belonging to Anybody do not possess any key and therefore are not able to perform any operation that is restricted to one of the roles which are explicitely excluded from the role Anybody.	

Subject	Nobody	Nobody
Info	Any subject that does not belong to one of the roles Admin, AppMgr, DelAppMgr, AppUser, AppChangeUser, AppRollUser, OrigKeyUser or Anybody, belongs to the role Nobody. Due to the definition of Anybody, the set of all subjects belonging to the role Nobody is the empty set.	

The objects are

Object	PICCLevelData	PICC Level Data
Info	The PICC level is the lowest level of the MIFARE DESFire Software (PICC level, Application level, File level). On the PICC level Application and DelApplication can be created or deleted. Hence to the PICCLevelData belong Application and DelApplication .	
Operation	Modify	Modify attributes of PICCLevelData.
Operation	Freeze	Freeze attributes of PICCLevelData.PICCKeySettings.
Attribute	PICCKeySettings	Generic PICC key settings.

Object	Application	Application
Info	The card can store a number of Application. An Application can store a number of File.	
Operation	Modify	Modify attribute Application.AppKeySettings.
Operation	Freeze	Freeze attribute Application.AppKeySettings.
Operation	Create	Create an Application.
Operation	Delete	Delete an Application.
Operation	Select	Select an Application.
Attribute	AppKeySettings	Generic application key settings.

Object	DelApplication	Delegated Application
Info	The card can store a number of DelApplication. After creation the DelApplication has the same attributes as a Application.	
Operation	Create	Create a DelApplication.
Operation	Delete	Delete a DelApplication.

Object	File	File
Info	An Application can store a number of File of different types.	
Operation	Create	Create a File.
Operation	Delete	Delete a File.
Operation	Freeze	Freeze attributes of File.
Operation	Read	Read operations accessing the content of a File.
Operation	Write	Write operations accessing the content of a File
Operation	ReadWrite	ReadWrite operations accessing the content of a File
Operation	Change	Change operation to change the attribute File.AccessRights
Attribute	AccessRights	Generic access rights for File.

Object	PICCMasterKey	PICC Master Key
Info	The Card Master Key.	
Operation	Change	Change the PICCMasterKey.
Operation	Freeze	Freeze the PICCMasterKey.

Object	PICCAppDefaultKey	PICC Application Default Key
Info	The Default Application Master Key and Application Keys that are used when an application is created and when a KeySet is initialized.	
Operation	Change	Change the PICCAppDefaultKey.

Object	PICCDAMAuthKey	PICC DAM Authentication Key
Info	Delegated Application Management Authentication Key	
Operation	Change	Change the PICCDAMAuthKey

Object	PICCDAMENCKey	PICC DAM Encryption Key
Info	Delegated Application Management Encryption Key to generate DAMENC.	
Operation	Change	Change the PICCDAMENCKey.

Object	PICCDAMMACKey	PICC DAM MAC Key
Info	Delegated Application Management MAC Key to generate DAMMAC.	
Operation	Change	Change the PICCDAMMACKey.

Object	OriginalityKey	Originality Key
Info	Key to check the originality of the card.	

Object	AppMasterKey	Application Master Key
Info	Application Master Key	
Operation	Change	Change the AppMasterKey
Operation	Freeze	Freeze the AppMasterKey

Object	AppChangeKey	Application Change Key
Info	Application Change Key	
Operation	Change	Change the AppChangeKey

Object	AppKey	Application Key
Info	Application Key	
Operation	Change	Change the AppKey.

Object	AppTransactionMACKey	Application Transaction MAC Key
Info	Application Transaction MAC Key	
Operation	Create	Create the AppTransactionMACKey.
Operation	Delete	Delete the AppTransactionMACKey.

Object	AppRollKey	Application Roll Keyset Key
Info	Application Roll Key Set Key	
Operation	Change	Change the AppRollKey.

Object	AppDAMDefaultKey	Application DAM Default Key
Info	Delegated Application Management Default Authentication Key	

Object	KeySet	Key Set
Info	AppKeys are grouped into KeySets.	
Operation	Roll	Roll the KeySet.

Note that subjects are authorized by cryptographic keys. These keys are considered as authentication data and not as security attributes of the subjects. The card has a card master key PICCMasterKey. Every Application has an AppMasterKey and a variable number of AppKeys organized in KeySet used for operations on Files (all these keys are called Application Keys). The Application Keys and Key Sets within an application are numbered.

The TOE shall meet the requirements ”Security Roles (FMT_SMR.1[DF])” as specified below.

FMT_SMR.1[DF]	**Security Roles**
Hierarchical-To	No other components.
Dependencies	FIA_UID.1 Timing of identification
FMT_SMR.1.1[DF]	The TSF shall maintain the roles *Admin*, *AppMgr*, *DelAppMgr*, *AppUser*, *AppChangeUser*, *AppRollUser*, *OrigKeyUser* and *Anybody*.
FMT_SMR.1.2[DF]	The TSF shall be able to associate users with roles.

The TOE shall meet the requirements ”Subset Access Control (FDP_ACC.1[DF])” as specified below.

FDP_ACC.1[DF]	**Subset Access Control**
Hierarchical-To	No other components.

Dependencies	FDP_ACF.1 Security attribute based access control.
FDP_ACC.1.1[DF]	The TSF shall enforce the *DESFire Access Control Policy* on *all subjects, objects, operations and attributes defined by the DESFire Access Control Policy*.

The TOE shall meet the requirements "Security Attribute Based Access Control (FDP_ACF.1[DF])" as specified below.

FDP_ACF.1[DF]	**Security Attribute Based Access Control**
Hierarchical-To	No other components.
Dependencies	FDP_ACC.1 Subset access control, FMT_MSA.3 Static attribute initialization
FDP_ACF.1.1[DF]	The TSF shall enforce the *DESFire Access Control Policy* to objects based on the following: *all subjects, objects and attributes*.
FDP_ACF.1.2[DF]	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:

DF_ACP_ACF1_21 *The Admin is allowed to perform Application.Create and Application.Delete.*

DF_ACP_ACF1_22 *The Admin is allowed to perform DelApplication.Delete.*

DF_ACP_ACF1_23 *The AppMgr is allowed to perform File.Create and File.Delete.*

DF_ACP_ACF1_24 *The DelAppMgr is allowed to perform DelApplication.Create with valid DAMMAC and valid DAMENC.*

FDP_ACF.1.3[DF] The TSF shall explicitly authorize access of subjects to objects based on the following additional rules:

DF_ACP_ACF1_31 *The AppMgr is allowed to Application.Delete if the attribute PICCLevelData.PICCKeySettings grant this right.*

DF_ACP_ACF1_32 *The AppUser is allowed to perfrom File.Read or File.Write or File.ReadWrite or File.Change on File if the File.AccessRights grant these rights.*

DF_ACP_ACF1_33 *The Anybody is allowed to perform Application.Create if the PICCLevelData.PICCKeySettings grant this right.*

DF_ACP_ACF1_34 *The Anybody is allowed to perform File.Create and File.Delete if the Application.AppKeySettings grant these rights.*

DF_ACP_ACF1_35 *Anybody is allowed to perform File.Read or File.Write or File.ReadWrite or File.Change if the File.AccessRights grant these rights.*

FDP_ACF.1.4[DF] The TSF shall explicitly deny access of subjects to objects based on the following additional rules:

DF_ACP_ACF1_41 *No one but Nobody is allowed to perform File.Read or File.Write or File.ReadWrite or File.Change if theFile.AccessRights do not grant this right.*

DF_ACP_ACF1_42 *OrigKeyUser is not allowed to perform any operation on objects.*

DF_ACP_ACF1_43 *No one but Nobody is allowed to perform any operation on OriginalityKey.*

The TOE shall meet the requirements ”Static Attribute Initialization (FMT_MSA.3[DF])” as specified below.

FMT_MSA.3[DF]	**Static Attribute Initialization**
Hierarchical-To	No other components.
Dependencies	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles
FMT_MSA.3.1[DF]	The TSF shall enforce the *DESFire Access Control Policy* to provide *permissive* default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2[DF]	The TSF shall allow *no one but Nobody* to specify alternative initial values to override the default values when an object or information is created.
Application Note:	The only initial attributes are the card attributes. All other attributes have to be defined at the same time the respective object is created.

The TOE shall meet the requirements ”Management of Security Attributes (FMT_MSA.1[DF])” as specified below.

FMT_MSA.1[DF]	**Management of Security Attributes**
Hierarchical-To	No other components.
Dependencies	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
FMT_MSA.1.1[DF]	The TSF shall enforce the *DESFire Access Control Policy* to restrict the ability to *modify or freeze and change* the security attributes *of the objects PICCLevelData, Application and the security attribute File.AccessRights* to *the Admin, AppMgr and AppChangeUser respectively*.
Refinement:	The detailed management abilities are:
DF_ACP_MSA1_11	*Only the Admin is allowed to perform PICCLevelData.Modify or PICCLevelData.Freeze on PICCLevelData.PICCKeySettings.*
DF_ACP_MSA1_12	*Only the AppMgr is allowed to perform Application.Modify or Application.Freeze on Application.AppKeySettings.*
DF_ACP_MSA1_13	*The AppChangeUser is allowed to perform File.Freeze on File.AccessRights.*

The TOE shall meet the requirements ”Management of TSF Data (FMT_MTD.1[DF])” as specified below.

FMT_MTD.1[DF]	**Management of TSF Data**
Hierarchical-To	No other components.
Dependencies	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
FMT_MTD.1.1[DF]	The TSF shall restrict the ability to *perform PICCMasterKey.Change, PICCMasterKey.Freeze, PICCAppDefaultKey.Change, AppMasterKey.Change, AppMasterKey.Freeze, AppChangeKey.Change* to *the Admin, AppMgr and AppUser.*

Refinement: The detailed management abilities are:

DF_ACP_MTD1_11 *Only the Admin is allowed to perform PICCMasterKey.Change or PICCMasterKey.Freeze.*

DF_ACP_MTD1_12 *The Admin is allowed to perform PICCAppDefaultKey.Change.*

DF_ACP_MTD1_13 *The Admin is allowed to perform PICCDAMAuthKey.Change.*

DF_ACP_MTD1_14 *The Admin is allowed to perform PICCDAMENCKey.Change.*

DF_ACP_MTD1_15 *The Admin is allowed to perform PICCDAMMACKey.Change.*

DF_ACP_MTD1_17 *The AppMgr is allowed to perform AppMasterKey.Change and AppMasterKey.Freeze.*

DF_ACP_MTD1_18 *The AppMgr is allowed to perform AppChangeKey.Change.*

DF_ACP_MTD1_19 *The AppMgr is allowed to perform AppKey.Change.*

DF_ACP_MTD1_1A *The AppMgr is allowed to perform AppRollKey.Change.*

DF_ACP_MTD1_1B *The AppMgr is allowed to perform AppTransactionMACKey.Create and AppTransactionMACKey.Delete.*

DF_ACP_MTD1_1C *The AppChangeUser is allowed to perform and AppChangeKey.Change.*

DF_ACP_MTD1_1D *The AppChangeUser is allowed to perform AppKey.Change.*

DF_ACP_MTD1_1E *The AppUser is allowed to perform AppKey.Change on AppKey if Application.AppKeySettings grant this right.*

DF_ACP_MTD1_1F *The AppUser is allowed to perform AppTransactionMACKey.Create and AppTransactionMACKey.Delete on AppTransactionMACKey if Application.AppKeySettings grant this right.*

DF_ACP_MTD1_10 *The AppRollUser is allowed to perform KeySet.Roll.*

The TOE shall meet the requirements ”Specification of Management Functions (FMT_SMF.1[DF])” as specified below.

FMT_SMF.1[DF]	**Specification of Management Functions**
Hierarchical-To	No other components.
Dependencies	No dependencies.
FMT_SMF.1.1[DF]	The TSF shall be capable of performing the following security management functions:

- *Authenticate a user,*
- *Invalidating the current authentication state based on the functions: Selecting an application or the card, Changing the key corresponding to the current authentication, Occurence of any error during the execution of a command, starting a new authentication, Rolling key set, Failed Proximity Check, Deleting an Application as AppMgr; Reset;*
- *Changing a security attribute*
- *rolling the keyset*
- *Creating or deleting an application, a delegated application or a file*
- *Selection of the Virtual Card*

The TOE shall meet the requirements ”Import of user data with security attributes (FDP_ITC.2[DF])” as specified below.

FDP_ITC.2[DF]	**Import of user data with security attributes**
Hierarchical-To	No other components.
Dependencies	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path] FPT_TDC.1 Inter-TSF basic TSF data consistency
FDP_ITC.2.1[DF]	The TSF shall enforce the *DESFire Access Control Policy* when importing user data, controlled under the SFP, from outside of the TOE.
FDP_ITC.2.2[DF]	The TSF shall use the security attributes associated with the imported user data.
FDP_ITC.2.3[DF]	The TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the user data received.
FDP_ITC.2.4[DF]	The TSF shall ensure that interpretation of the security attributes of the imported user data is as intended by the source of the user data.
FDP_ITC.2.5[DF]	The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: *no additional rules*.

6.1.2.2 Implications of the DESFire Access Control Policy

The DESFire Access Control Policy has some implications, that can be drawn from the policy and that are essential parts of the TOE security functions.

- The TOE end-user does normally not belong to the group of authorised users (Admin, AppMgr, DelAppMgr, AppUser), but regarded as Anybody by the TOE. This means that the TOE cannot determine if it is used by its intended end-user (in other words: it cannot determine if the current card holder is the owner of the card).
- The Admin can have the exclusive right to create and delete Applications on the card, however he can also grant this privilege to Anybody. In the case of DelApplications the Admin can grant this privilege to the AppMgr. Additionally, changing the PICCLevelData is reserved for the Admin. AppKeys, at delivery time should be personalized to a preliminary, temporary key only known to the Admin and the AppMgr.
- At Application personalization time, the AppMgr uses the preliminary AppKey in order to personalize the AppKeys, whereas all keys, except the AppMasterKey, can be personalized to a preliminary, temporary key only known to the AppMgr and the AppUser. Furthermore, the AppMgr has the right to create Files within his Application scope.

6.1.3 Additional SFRs regrading confidentiality, authentication and integrity

The TOE shall meet the requirements ”Cryptographic Operation (DES) (FCS_COP.1[DF-DES])” as specified below.

FCS_COP.1[DF-DES]	**Cryptographic Operation (DES)**

Hierarchical-To	No other components.
Dependencies	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation], FCS_CKM.4 Cryptographic key destruction
FCS_COP.1.1[DF-DES]	The TSF shall perform *encryption and decryption used for authentication* in accordance with the specified cryptographic algorithm *Triple-DES in one of the following modes of operation: CBC and 3-key Triple-DES* and cryptographic key sizes *168 bit* that meet the following *standards*: • *FIPS PUB 46-3 FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION DATA ENCRYPTION STANDARD (DES) Reaffirmed 1999 October 25, keying options 1 and 2*

The TOE shall meet the requirements ”Cryptographic Operation (AES) (FCS_COP.1[DF-AES])” as specified below.

FCS_COP.1[DF-AES]	**Cryptographic Operation (AES)**
Hierarchical-To	No other components.
Dependencies	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation], FCS_CKM.4 Cryptographic key destruction
FCS_COP.1.1[DF-AES]	The TSF shall perform *encryption and decryption and cipher based MAC for authentication and communication* in accordance with the specified cryptographic algorithm *Advanced Encryption Standard AES in one of the following modes of operation: CBC, CMAC* and a cryptographic key size *of 128 bits* that meet the following *standards*: • *FIPS Publication 197, Advanced Encryption Standard (AES),* • *NIST Special Publication 800- 38A, 2001 (CBC mode) [11] and* • *NIST Special Publication 800-38B (CMAC mode) [12]*
Refinement:	For the MIFARE DESFire EV1 secure messaging the TOE uses the cryptographic algorithm for CMAC according to NIST Special Publication 800-38B (CMAC mode) [12] with the following modification: The TOE does not use the standard zero byte IV instead it uses an IV defined by the previous cryptographic operation (chaining mode).

The TOE shall meet the requirements ”User identification before any Action (FIA_UID.2[DF])” as specified below.

FIA_UID.2[DF]	**User identification before any Action**
Hierarchical-To	FIA_UID.1
Dependencies	No dependencies.
FIA_UID.2.1[DF]	The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

Application Note: Identification of a user is performed upon an authentication request based on the currently selected context and the key number. For example, if an authentication request for key number 0 is issued after selecting a specific application, the user is identified as the Application Manager of the respective application. Before any authentication request is issued the user is identified as ”Everybody”.

The TOE shall meet the requirements ”User Authentication before any Action (FIA_UAU.2[DF])” as specified below.

FIA_UAU.2[DF]	**User Authentication before any Action**
Hierarchical-To	FIA_UAU.1
Dependencies	FIA_UID.1 Timing of identification
FIA_UAU.2.1[DF]	The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

The TOE shall meet the requirements ”Multiple Authentication Mechanisms (FIA_UAU.5[DF])” as specified below.

FIA_UAU.5[DF]	**Multiple Authentication Mechanisms**
Hierarchical-To	No other components.
Dependencies	No dependencies.
FIA_UAU.5.1[DF]	The TSF shall provide *'none' and cryptographic authentication* to support user authentication.
FIA_UAU.5.2[DF]	The TSF shall authenticate any user's claimed identity according to the *following rules*: • *The 'none' authentication is performed with anyone who communicates with the TOE without issuing an explicit authentication request. The 'none' authentication implicitly and solely authorizes the 'Everybody' subject.* • *The cryptographic authentication is used to authorise the Administrator, Application Manager, Delegated Application Manager and Application User.*

Refinement: For the applied cryptographic operation please refer to FCS_COP.1[DF-AES] and FCS_COP.1[DF-DES]

The TOE shall meet the requirements ”Trusted Path (FTP_TRP.1[DF])” as specified below.

FTP_TRP.1[DF]	**Trusted Path**
Hierarchical-To	No other components.
Dependencies	No dependencies.
FTP_TRP.1.1[DF]	The TSF shall provide a communication path between itself and *remote* users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from *modification and disclosure or only modification*.
FTP_TRP.1.2[DF]	The TSF shall permit *remote users* to initiate communication via the trusted path.
FTP_TRP.1.3[DF]	The TSF shall require the use of the trusted path for *authentication requests with 3 key Triple-DES or AES, confidentiality and/or integrity verification for data transfers protected with AES based on a setting in the file attributes*.

The TOE shall meet the requirements ”Cryptographic Key Destruction (FCS_CKM.4[DF])” as specified below.

FCS_CKM.4[DF]	**Cryptographic Key Destruction**
Hierarchical-To	No other components.
Dependencies	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic Key Generation]
FCS_CKM.4.1[DF]	The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method *overwriting* that meets the following: *none.*

The TOE shall meet the requirements ”Inter-TSF Basic TSF Data Consistency (FPT_TDC.1[DF])” as specified below.

FPT_TDC.1[DF]	**Inter-TSF Basic TSF Data Consistency**
Hierarchical-To	No other components.
Dependencies	No dependencies.
FPT_TDC.1.1[DF]	The TSF shall provide the capability to consistently interpret *data files and values* when shared between the TSF and another trusted IT product.
FPT_TDC.1.2[DF]	The TSF shall use *the rules: data files or values can only be modified by their dedicated type-specific operations honouring the type-specific boundaries* when interpreting the TSF data from another trusted IT product.

6.1.4 Additional SFRs regrading the robustness

The TOE shall meet the requirements ”Basic rollback (FDP_ROL.1[DF])” as specified below.

FDP_ROL.1[DF]	**Basic rollback**
Hierarchical-To	No other components.
Dependencies	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]
FDP_ROL.1.1[DF]	The TSF shall enforce Access Control Policy to permit the rollback of *the operations that modify the value or data file objects* on *the backup files*.
FDP_ROL.1.2[DF]	The TSF shall permit operations to be rolled back within *the scope of the current transaction, which is defined by the following limitative events: chip reset, select command, deselect command, explicit commit, explicit abort, command failure*.

The TOE shall meet the requirements ”Replay detection (FPT_RPL.1[DF])” as specified below.

FPT_RPL.1[DF]	**Replay detection**
Hierarchical-To	No other components.
Dependencies	No dependencies.
FPT_RPL.1.1[DF]	The TSF shall detect replay for the following entities: *authentication requests with 3-key Triple-DES or AES, confidentiality and/or data integrity verification for data transfers protected with AES and based on a setting in the file attributes*.

FPT_RPL.1.2[DF]	The TSF shall perform *rejection of the request* when replay is detected.

The TOE shall meet the requirements ”Unlinkability (FPR_UNL.1[DF])” as specified below.

FPR_UNL.1[DF]	**Unlinkability**
Hierarchical-To	No other components.
Dependencies	No dependencies.
FPR_UNL.1.1[DF]	The TSF shall ensure that *unauthorised subjects other than the card holder* are unable to determine whether *any operation of the TOE were caused by the same user*.

6.2 Security Assurance Requirements

Table 6.28 below lists all security assurance components that are valid for this Security Target. With two exceptions these security assurance components are required by EAL5 (see section 2.3) or by the Protection Profile. The exception are the components ASE_TSS.2 and ALC_FLR.1 which are chosen as an augmentation in this Security Target. ASE_TSS.2 is chosen to give architectural information on the security functionality of the TOE. ALC_FLR.1 is chosen to give assurance that the TOE will be maintained and supported in the future.

The refinements of the Protection Profile that must be adapted for EAL5 are described in section 6.2.1.

Name	Title
ADV_ARC.1	Security architecture description
ADV_FSP.5	Complete semi-formal functional specification with additional error information
ADV_IMP.1	Implementation representation of the TSF
ADV_INT.2	Well-structured internals
ADV_TDS.4	Semiformal modular design
AGD_OPE.1	Operational user guidance
AGD_PRE.1	Preparative procedures
ALC_CMC.4	Production support, acceptance procedures and automation
ALC_CMS.5	Development tools CM coverage
ALC_DEL.1	Delivery procedures
ALC_DVS.2	Sufficiency of security measures
ALC_FLR.1	Basic flaw remediation
ALC_LCD.1	Developer defined life-cycle model
ALC_TAT.2	Compliance with implementation standards
ASE_INT.1	ST introduction
ASE_CCL.1	Conformance claims
ASE_SPD.1	Security problem definition
ASE_OBJ.2	Security objectives

Name	Title
ASE_ECD.1	Extended components definition
ASE_REQ.2	Derived security requirements
ASE_TSS.2	TOE summary specification with architectural design summary
ATE_COV.2	Analysis of coverage
ATE_DPT.3	Testing: modular design
ATE_FUN.1	Functional testing
ATE_IND.2	Independent testing - sample
AVA_VAN.5	Advanced methodical vulnerability analysis

Tab. 6.28: Security Assurance Requirements

6.2.1 Refinements of the TOE Security Assurance Requirements

In compliance to Application Note 23 in the Protection Profile, this Security Target has to conform to all refinements of the security assurance requirements in the Protection Profile. Because the refinements in the Protection Profile are defined for the security assurance components of EAL4 (augmented by ALC_DVS.2 and AVA_VAN.5), some refinements have to be applied to assurance components of the higher level EAL5 stated in the Security Target.

Table 6.29 lists the influences of the refinements of the Protection Profile on the Security Target. Most of the refined security assurance components have the same level in both documents (Protection Profile and Security Target). The following two subsections apply the refinements to ALC_CMS.5 and ADV_FSP.5, which are different between the Protection Profile and the Security Target.

SAR in PP [13]	Effect on Security Target
ALC_DEL.1	Same as in PP, refinement valid without change
ALC_DVS.2	Same as in PP, refinement valid without change
ALC_CMS.4	ALC_CMS.5, refinements valid without change
ALC_CMC.4	Same as in PP, refinement valid without change
ADV_ARC.1	Same as in PP, refinement valid without change
ADV_FSP.4	ADV_FSP.5, refinements have to be adapted
ADV_IMP.1	Same as in PP, refinement valid without change
ATE_COV.2	Same as in PP, refinement valid without change
AGD_OPE.1	Same as in PP, refinement valid without change
AGD_PRE.1	Same as in PP, refinement valid without change
AVA_VAN.5	Same as in PP, refinement valid without change

Tab. 6.29: SARs refined in the PP [13] and their effect on this ST

6.2.1.1 Refinements regarding CM scope (ALC_CMS)

This Security Target requires a higher evaluation level for the CC family ALC_CMS, namely ALC_CMS.5 instead of ALC_CMS.4. The refinement of the Protection Profile regarding ALC_CMS.4 is a clarification of the configuration item ”TOE implementation representation”. Since in ALC_CMS.5, the content and presentation of evidence element ALC_CMS.5.1C only adds a further configuration item to the list of items to be tracked by the CM system, the refinement can be applied without changes.

The refinement of the configuration item ”TOE implementation representation” of ALC_CMS.4 can be found in section 6.2.1.3 of the Protection Profile and is not cited here.

6.2.1.2 Refinements regarding ADV_FSP

This Security Target requires a higher evaluation level for the CC family ADV_FSP, namely ADV_FSP.5 instead of ADV_FSP.4. The refinement of the Protection Profile regarding ADV_FSP.4 is concerned with the complete representation of the TSF, the purpose and method of use of all TSFI, 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 above items.

The higher level ADV_FSP.5 requires a Functional Specification in a ”semi-formal style” (ADV_FSP.5.2C). The component ADV_FSP.5 enlarges 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 refinements can be applied without changes and are valid for ADV_FSP.5. The refinement of the original component ADV_FSP.4 can be found in section 6.2.1.6 of the Protection Profile and is not cited here.

6.3 Security Requirements Rationale

6.3.1 Rationale for the Security Functional Requirements

Section 6.3.1 in the Protection Profile provides a rationale for the mapping between security functional requirements and security objectives defined in the Protection Profile. The mapping is reproduced in the following table.

SO	SFR
O.Leak-Inherent	FDP_ITT.1[HW] FDP_IFC.1 FPT_ITT.1[HW]
O.Phys-Probing	FDP_SDC.1[HW] FPT_PHP.3

SO	SFR
O.Malfunction	FPT_FLS.1 FRU_FLT.2
O.Phys-Manipulation	FDP_SDI.2[HW] FPT_PHP.3
O.Leak-Forced	FDP_ITT.1[HW] FDP_IFC.1 FPT_FLS.1 FPT_ITT.1[HW] FPT_PHP.3 FRU_FLT.2
O.Abuse-Func	FDP_ITT.1[HW] FDP_IFC.1 FMT_LIM.1[HW] FMT_LIM.2[HW] FPT_FLS.1 FPT_ITT.1[HW] FPT_PHP.3 FRU_FLT.2
O.Identification	FAU_SAS.1[HW]
O.RND	FCS_RNG.1[HW] FDP_ITT.1[HW] FDP_IFC.1 FPT_FLS.1 FPT_ITT.1[HW] FPT_PHP.3 FRU_FLT.2 FCS_RNG.1[DET]

Tab. 6.30: Security Functional Requirements vs. Security Objectives (PP)

The Security Target additionally defines the SFRs for the TOE that are listed in Table 6.31. In addition Security Requirements for the Environment are defined. The following table gives an overview, how the requirements are combined to meet the security objectives.

SO	SFR
O.Access-Control	FCS_CKM.4[DF] FDP_ACC.1[DF] FDP_ACF.1[DF] FDP_ITC.2[DF] FMT_MSA.1[DF] FMT_MSA.3[DF]

SO	SFR
	FMT_MTD.1[DF] FMT_SMF.1[DF] FMT_SMR.1[DF]
O.Authentication	FCS_COP.1[DF-DES] FCS_COP.1[DF-AES] FIA_UID.2[DF] FIA_UAU.2[DF] FIA_UAU.5[DF] FMT_SMF.1[DF] FPT_RPL.1[DF] FTP_TRP.1[DF]
O.Encryption	FCS_CKM.4[DF] FCS_COP.1[DF-AES] FTP_TRP.1[DF]
O.MAC	FCS_CKM.4[DF] FCS_COP.1[DF-AES] FPT_RPL.1[DF] FTP_TRP.1[DF]
O.Type_Consistency	FPT_TDC.1[DF]
O.Transaction	FDP_ROL.1[DF]
O.No-Trace	FPR_UNL.1[DF]

Tab. 6.31: Security Functional Requirements vs. Security Objectives (ST)

Justification related to "Access Control (O.Access-Control)"
The SFR FMT_SMR.1[DF] defines the roles of the Access Control Policy. The SFR FDP_ACC.1[DF] and FDP_ACF.1[DF] define the rules and FMT_MSA.3[DF] and FMT_MSA.1[DF] the attributes that the access control is based on. FMT_MTD.1[DF] provides the rules for the management of the authentication data. The management functions are defined by FMT_SMF.1[DF]. Since the TOE stores data on behalf of the authorised subjects import of user data with security attributes is defined by FDP_ITC.2[DF]. Since cryptographic keys are used for authentication (refer to O.Authentication), these keys have to be removed if they are no longer needed for the access control (i.e. an application is deleted). This is required by FCS_CKM.4[DF]. These nine SFR together provide an access control mechanism as required by the objective O.Access-Control.
Justification related to "Authentication (O.Authentication)"
The two SFR FCS_COP.1[DF-DES] and FCS_COP.1[DF-AES] require that the TOE provides the basic cryptographic algorithms that can be used to perform the authentication. The SFR FIA_UID.2[DF], FIA_UAU.2[DF] and FIA_UAU.5[DF] together define that users must be identified and authenticated before any action. The "none" authentication of FIA_UAU.5[DF] also ensures that a specific subject is identified and authenticated before an explicit authentication request is sent to the TOE. FMT_SMF.1[DF] defines security management functions the TSF

shall be capable to perform. FTP_TRP.1[DF] requires a trusted communication path between the TOE and remote users, FTP_TRP.1.3[DF] especially requires "authentication requests". Together with FPT_RPL.1[DF] which requires a replay detection for these authentication requests the eight SFR fulfill the objective O.Authentication.

Justification related to "Confidential Communication (O.Encryption)"

The SFR FCS_COP.1[DF-AES] requires that the TOE provides the basic cryptographic algorithm AES that can be used to protect the communication by encryption. FTP_TRP.1[DF] requires a trusted communication path between the TOE and remote users, FTP_TRP.1.3[DF] especially requires "confidentiality and/or data integrity verification for data transfers protected with AES and based on a setting in the file attributes". FCS_CKM.4[DF] requires that cryptographic keys used for encryption have to be removed after usage. These three SFR fulfill the objective O.Encryption.

Justification related to "Integrity-protected Communication (O.MAC)"

The SFR FCS_COP.1[DF-AES] requires that the TOE provides the basic cryptographic algorithms that can be used to compute a MAC which can protect the integrity of the communication. FTP_TRP.1[DF] requires a trusted communication path between the TOE and remote users, FTP_TRP.1.3[DF] especially requires "'confidentiality and/or data integrity verification for data transfers on request of the file owner"'. FCS_CKM.4[DF] requires that cryptographic keys used for MAC operations have to be removed after usage. Together with FPT_RPL.1[DF] which requires a replay detection for these data transfers the four SFR fulfill the objective O.MAC.

Justification related to "Data type consistency (O.Type_Consistency)"

The SFR FPT_TDC.1[DF] requires the TOE to consistently interpret data files and values. The TOE will honor the respective file formats and boundaries (i.e. upper and lower limits, size limitations). This meets the objective O.Type_Consistency.

Justification related to "Transaction mechanism (O.Transaction)"

The SFR FDP_ROL.1[DF] requires the possibility to rollback a set of modifying operations on backup files in total. The set of operations is defined by the scope of the transaction, which is itself limited by some boundary events. This fulfils the objective O.Transaction.

Justification related to "Preventing Traceability (O.No-Trace)"

The SFR FPR_UNL.1[DF] requires that unauthorised subjects other than the card holder are unable to determine whether any operation of the TOE were caused by the same user. This meets the objective O.No-Trace.

6.3.2 Dependencies of Security Functional Requirements

The dependencies listed in the Protection Profile are independent of the additional dependencies listed in the table below. The dependencies of the Protection Profile are fulfilled within the Protection Profile and at least one dependency is considered to be satisfied. The following discussion demonstrates how the SFR dependencies (defined by Part 2 of the Common Criteria [3]) satisfy the requirements specified in section 6.1.

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

SFR	Dependencies	Fulfilled by Security Requirements in the ST
FAU_SAS.1[HW]	No dependencies.	No dependency
FCS_RNG.1[HW]	No dependencies.	No dependency
FCS_RNG.1[DET]	No dependencies.	No dependency
FDP_ITT.1[HW]	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]	Yes
FDP_IFC.1	FDP_IFF.1 Simple security attributes	See discussion in the PP
FDP_SDC.1[HW]	No dependencies.	No dependency
FDP_SDI.2[HW]	No dependencies.	No dependency
FMT_LIM.1[HW]	FMT_LIM.2 Limited availability.	Yes
FMT_LIM.2[HW]	FMT_LIM.1 Limited capabilities.	Yes
FPT_FLS.1	No dependencies.	No dependency
FPT_ITT.1[HW]	No dependencies.	No dependency
FPT_PHP.3	No dependencies.	No dependency
FRU_FLT.2	FPT_FLS.1 Failure with preservation of secure state.	Yes

Tab. 6.32: Dependencies of Security Functional Requirements (PP)

SFR	Dependencies	Fulfilled by Security Requirements in the ST
FCS_CKM.4[DF]	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic Key Generation]	Yes, by FDP_ITC.2[DF].
FCS_COP.1[DF-DES]	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation], FCS_CKM.4 Cryptographic key destruction	Yes, by FDP_ITC.2[DF]. Yes, by FCS_CKM.4[DF].

SFR	Dependencies	Fulfilled by Security Requirements in the ST
FCS_COP.1[DF-AES]	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation], FCS_CKM.4 Cryptographic key destruction	Yes, by FDP_ITC.2[DF]. Yes, by FCS_CKM.4[DF].
FDP_ACC.1[DF]	FDP_ACF.1 Security attribute based access control.	Yes, by FDP_ACF.1[DF].
FDP_ACF.1[DF]	FDP_ACC.1 Subset access control, FMT_MSA.3 Static attribute initialization	Yes, by FDP_ACC.1[DF]. Yes, by FMT_MSA.3[DF].
FDP_ITC.2[DF]	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path] FPT_TDC.1 Inter-TSF basic TSF data consistency	Yes, by FDP_ACC.1[DF]. Yes, by FTP_TRP.1[DF]. Yes, by FPT_TDC.1[DF].
FDP_ROL.1[DF]	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]	Yes, by FDP_ACC.1[DF].
FIA_UID.2[DF]	No dependencies.	No dependency
FIA_UAU.2[DF]	FIA_UID.1 Timing of identification	Yes, by FIA_UID.2[DF].
FIA_UAU.5[DF]	No dependencies.	No dependency
FMT_MSA.1[DF]	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	Yes, by FDP_ACC.1[DF]. Yes, by FMT_SMR.1[DF]. Yes, by FMT_SMF.1[DF].
FMT_MSA.3[DF]	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles	Yes, by FMT_MSA.1[DF]. Yes, by FMT_SMR.1[DF].
FMT_MTD.1[DF]	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	Yes, by FMT_SMR.1[DF]. Yes, by FMT_SMF.1[DF].
FMT_SMF.1[DF]	No dependencies.	No dependency

SFR	Dependencies	Fulfilled by Security Requirements in the ST
FMT_SMR.1[DF]	FIA_UID.1 Timing of identification	Yes, by FIA_UID.2[DF].
FPR_UNL.1[DF]	No dependencies.	No dependency
FPT_RPL.1[DF]	No dependencies.	No dependency
FPT_TDC.1[DF]	No dependencies.	No dependency
FTP_TRP.1[DF]	No dependencies.	No dependency

Tab. 6.33: Dependencies of Security Functional Requirements (Security Target)

6.3.3 Rationale for the Assurance Requirements

The selection of assurance components is based on the underlying Protection Profile. The Security Target uses the same augmentations as the Protection Profile, but chooses a higher assurance level. The level EAL5 is chosen in order to meet assurance expectations of access control applications and automatic fare collection systems. Additionally, the requirement of the Protection Profile to choose at least EAL4 is fulfilled.

The rationale for the augmentations is the same as in the Protection Profile. The assurance level EAL5 is an elaborated pre-defined level of the CC, part 3 [4]. 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 EAL5. Therefore, these components add additional assurance to EAL5, but the mutual support of the requirements is still guaranteed.

6.3.4 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 access control function used to implement the Access Control Policy. The security objectives defined in the Protection Profile can be seen as "lowlevel protection" objectives, while the additional security objectives defined in this Security Target are "high-level protection" objectives. For example, O.Encryption states that the communication can be protected by encryption. While this ensures the rather high-level goal that the communication can not be eavesdropped, the overall goal that the communication is confidential is ensured with the help of the Protection

Profile objective that prevent attacks on the key and the cryptographic implementation like probing or fault injection attacks.

7 TOE Summary Specification

7.1 Portions of the TOE Security Functionality

The TSF directly corresponds to the TOE security functional requirements defined in Section 6.

The following portions of security functionality are applicable to the phases 4 to 7.

Remark 2. Parts of the security functionality are configured at the end of phase 3 and the whole security functionality is already active during the delivery from phase 3 to phase 4.

The TOE comprises additional features that are not listed as security functionality in the following. They do not provide a complete portion of the security functionality by themselves but they can be used to support a portion of the security functionality implemented by the MIFARE DESFire Software, as for example the CRC calculation for the control of data integrity.

The TSF described in the following is split into Security Services and Security Features.

7.1.1 Security Services

SS.AUTH **Authentication**

The TOE provides an authentication mechanism to separate authorised subjects from unauthorised subjects. The authentication of subjects is performed by a cryptographic challenge-response. The TOE supports the cryptographic algorithms 3-key Triple-DES and 128-bit AES; for 3-key Triple-DES according to FIPS PUB 46-3 [7] and for AES according to FIPS PUB 197 [6]. The authentication mechanisms are implemented using the cryptographic coprocessors and the hardware random number generator provided by the hardware platform. The authentication mechanisms are protected against attacks like e.g. replay.

SS.AUTH identifies the user to be authenticated by the currently selected context (card or specific application, chosen by a "select" command) and the key number indicated in the authentication request. By default and before any authentication request SS.AUTH identifies and authenticates the role Anybody. The roles Admin, AppMgr, DelAppMgr, AppUser, AppChangeUser, AppRollUser and OrigKeyUser are authenticated during the authentication request by the knowledge of the respective cryptographic key.

The authentication state is remembered by SS.AUTH and the authentication need not to be performed again as long as none of the following events occur: Issue of a "select" command, occurrence of any error during the processing of a command, change of the key, or key set that was used for authentication and reset (any cause, either internal or external reset). These events will reset the authentication state to the default (Anybody).

Additionally, if the AppMgr deletes his Application the authentication state will be reset as an implication.

Remark 3. Note that the TOE does also allow Single-DES and 2-key Triple-DES, but this shall not be used in the evaluated product. The TOE supports a backward compatible DES authentication in addition to the standard DES authentication. The backward compatible DES authentication shall not be used in the evaluated product.

SS.ACC_CTRL **Access Control**

SS.ACC_CTRL provides an access control mechanism to the Objects and Security Attributes that are part of the DESFire Access Control Policy. The access control mechanism assigns subjects - (possibly multiple) AppUsers - to 4 different groups of operations on Files. The operations are File.Read, File.Write, File.ReadWrite and File.Change. One subject can be assigned to each group of File operations. The special subjects Anybody and Nobody can also be assigned. For Files the operations furthermore are File.Create and File.Delete. These operations can be assigned to the AppMgr or to Anybody. The assignment is stored in the Application attributes. If a File is created the File attributes must be supplied with the File.Create request.
For the Application the operations are Application.Create and Application.Delete. These operations can be assigned to the Admin or to Anybody. The assignment is stored in the PICCLevelData.PICCKeySettings. Additionaly, the Admin can delegate Application creation to a DelAppMgr by the use of DelApplication. If an Application is created the attributes Application.AppKeySettings must be supplied with the Application.Create request. A Application.Delete operation will securely delete all application keys by overwriting them with random values.

SS.ACC_CTRL also controls access to the Security Attributes and the authentication data. The Card attributes and the PICCMasterKey can only be changed by the Admin, as long as the Admin does not freeze the PICCLevelData.PICCKeySettings or freezes the PICCMasterKey. The Application attributes and AppMasterKeys can be changed by the AppMgr, as long as the AppMgr does not freeze the Application.AppKeySettings or the AppMasterKey. Additionally the AppMgr can change the AppKeys and decide if the AppUser can change their AppKeys or not. For Files, the attributes can be changed by the subject that has the File.AccessRights to perform the operation File.Change. SS.ACC_CTRL allows the Admin to specify a PICCAppDefaultKey and AppKeys that will be used when an Application is created.

The OrigKeyUser is not allowed to perform any operation on objects, but with a successful authentication he can prove the authenticity of the Security IC.

Finally SS.ACC_CTRL ensures the type consistency of the File types stored by the TOE. It ensures that values can not over- or underflow. Furthermore size limitations of Files are obeyed by SS.ACC_CTRL.

SS.ENCRYPTION **Encryption**

The TSF SS.ENCRYPTION provides a mechanism to protect the communication against eavesdropping. In order to do this the communication can be encrypted. The encryption is requested by the file owner (i.e. the subject that has the right to ”change attribute” for a file) by setting an option in the file attributes.

The encryption algorithm is the same as the one used during authentication for the session, however SS.ENCRYPTION only supports the AES algorithm, therefore it is bound to authentications with this algorithm. Note that the TSF SS.ENCRYPTION is active after authentication performed with SS.AUTH.

SS.ENCRYPTION also adds data to the communication stream that enables the terminal to detect integrity violations, replay attacks or man-in-the-middle attacks.

If an encrypted communication is requested, SS.ENCRYPTION also verifies the data sent by the terminal and returns an error code if such an attack is detected. The detection mechanism covers all frames exchanged between the terminal and the card up to the current encrypted frame. Therefore SS.ENCRYPTION can detect any injected/modified frame in the communication before the transfer of the encrypted frame.

SS.MAC **Message Authentication Code**

The TSF SS.MAC provides a mechanism for integrity protection, replay attack protection and protection against man-in-the-middle attacks on the communication path. The integrity protection is requested by the File owner (i.e. the subject that has the right to perform File.Change for a File) by setting an option in the attribute File.AccessRights.

SS.MAC adds data to the communication stream that enables both the TOE and the terminal to detect integrity violations, replay attacks or man-in-the-middle attacks using the cryptographic algorithm 128-bit AES CMAC, see [12]. Note that SS.MAC only supports the AES algorithm. If an integrity protected communication is requested, SS.MAC verifies the data sent by the terminal and returns an error code if such an attack is detected. The detection mechanism covers all frames exchanged between the terminal and the TOE up to the current integrity protected frame. Therefore SS.MAC can detect any injected/modified frame in the communication before the transfer of the integrity protected frame.

SS.TRANSACTION **Transaction**

The transaction mechanism implemented by SS.TRANSACTION ensures that either all or none of the (modifying) commands within a transaction are performed. The transaction mechanism is active for backup data files, values, linear record files and cyclic record files, it is not active for standard data files. All file types with the exception of ”standard data files” are called ”backup files” in the following.

SS.TRANSACTION is always active for the respective file types. This means that for every modifying operation with a backup file an explicit commit request must be issued in order to let the modifications take effect. Note that it is possible by the use of the shared application feature, that Files in up to 2 Applications can be updated within

one transaction.

Several reasons will abort a transaction: These are the explicit abort request, chip reset, a "select" command, a deselect command, a roll key set command, a create or delete transaction MAC file command, a delete or format application command, a format card command or any failure of a command.

SS.TRANSACTION_MAC **Transaction Message Authentication Code**

SS.TRANSACTION_MAC ensures that a MAC is calculated over a commited transaction with the dedicated AppTransactionMACKey, wich exists per Application. Note that a commited transaction consists of a sequence of operations on the TOE.

SS.TRANSACTION_MAC is a security service on application level, which can be activated per Application. This is done by creating a so called "'TransactionMAC file"' and defining a AppTransactionMACKey. SS.TRANSACTION_MAC provides a service to AppUsers and AppMgrs or Admins. SS.TRANSACTION_MAC helps AppUsers to prove the authenticity of committed transactions on the TOE towards the AppMgr or Admin.

The transaction MAC, calculated by SS.TRANSACTION_MAC, also involves a Transaction MAC Counter maintained by the TOE, which helps the AppMgr or Admin to detect replay by the AppUser.

SS.NO_TRACE **Preventing Traceability**

SS.NO_TRACE provides an option to use a random ID during the ISO14443 anti-collision sequence [17]. If this option is set, the TOE does not send its UID, but generates a new random ID number during every power-on sequence. By this the card cannot be traced any more by simply retrieving its UID.

Card specific information suitable to identify single end-users comprises the UID. All card specific information can be read out only by the Admin, AppMgr and AppUser if the option for the random UID is set. Setting this option is restricted to the Admin.
SS.NO_TRACE further provides an option to use the Virtual Card Architecture. This allows using the TOE in a complex environment where multiple Virtual Cards are stored in one physical object, however the TOE does support only one virtual card.

Remark 4. Note that SS.NO_TRACE protects the card specific data. In order to prevent traceability at all the authorised subjects have to make use of the access control mechanism implemented by SS.ACC_CTRL.

By using SS.NO_TRACE and SS.ACC_CTRL it can be ensured that no unauthorised subject can gain information about the end-user that allows to identify the end-user. As a consequence this does not allow to trace the end-user, e.g. by setting up a terminal controlled by an attacker.

7.1.2 Security Features

SF.OPC **Control of Operating Conditions**

SF.OPC ensures the correct operation of the TOE (functions offered by the micro-controller including the standard CPU as well as the unified AES/Triple-DES co-processor, the memories, registers, I/O interfaces and the other system peripherals) during the execution of the IC Dedicated Support Software. This includes all specific security features of the TOE which are able to provide an active response.

The TOE ensures its correct operation and prevents any malfunction by means of three kinds of features:

Environmental Control: Set of security mechanisms that detect if the TOE runs out of the specified operation conditions. It needs to be assured that in operation mode all ambient conditions are within their specified limits. Sensors take over the role of measuring the ambient conditions and reacting in case of specification violation of one of the ambient parameters. If a sensor monitors a violation of the specified ambient conditions, a reset is triggered. Depending on the type of sensor the reset might be a security reset that decrements the error counter.

Execution Integrity Set of security mechanisms that detect if an execution of an operation has been manipulated. It needs to be assured that manipulations on operations are detected and trigger a reset that effects the error counter. Manipulating operations means the operation itself is attacked. On an abstract view this could mean that some kind of memory (e.g. register) has been attacked. On a more detailed view it can also mean that entire wires or gates are attacked. Executing integrity is achieved by means such as the following ones:

- validity checking of in- and output of security critical operations
- integrity protection of data, code and address path
- integrity protection of memories, data registers, key registers and control registers
- monitoring state machines
- integrity protection of sensor signals
- double calculations and checks

Integrity protection is achieved by various techniques, such as parity protection, redundant encoding and execution, monitoring, CRCs.

Availability Set of security mechanisms that take care that the availability of the TOEs functionality is limited if attacks occur. It needs to be assured that the detection of an attack results in secure state. This is achieved by the fact that any kind of attack or operation outside the operation conditions results in a reset where the TOE boots in the initial configuration. Depending in the kind of reset source the reset might also have an effect on the error counter. This is especially the case for integrity violations that cannot be unintended ones.

SF.PHY **Protection against Physical Manipulation**

The feature SF.PHY protects the TOE against manipulation of

(i) the hardware,

(ii) the IC Dedicated Software in the non-volatile memory, and

(iii) the application data in the RAM and EEPROM including the configuration data stored in EEPROM.

It also protects all data stored in the memories including User Data and TSF data against disclosure by physical probing when stored or while being processed by the TOE.

The TOE ensures its correct operation and prevents any malfunction by means of several kinds of features:

- **Layout Protection:** Set of security mechanisms that hamper reverse engineering of the IC, such as layout randomization, active and passive shielding, techniques to hide shielding, multilayer interconnection, wide bus widths and dummy routing.
- **Code- & Datapath Integrity Protection:** Set of security mechanisms that ensure that manipulations on data or code stored and transmitted from respectively to the CPU are detected with high probability. This includes integrity protection of the whole code and data path including CPU internals. Integrity verification is always done before the according data is processed via e.g. an ALU operation.
- **Memory Integrity Protection:** Set of security mechanisms that ensure that manipulations on memory content are detected with high probability. This includes integrity protection of memories and registers. EEPROM are additionally equipped with error correction codes, double read technology and anti-tearing.
- **Address Path Integrity Protection:** Set of security mechanisms that ensure that manipulations on the address path are detected with high probability.
- **Startup Integrity Protection:** Set of security mechanisms that detect integrity errors during startup (e.g. with respect to configuration data).
- **Redundant Encoding:** Set of security mechanisms that ensure that security critical flags and the according checks are kept with an according level of redundancy.
- **Code Integrity Protection:** Set of security mechanisms that detect if code has been manipulated.
- **Code- & Datapath Encryption:** Set of security mechanisms that ensure that code or data processed by the CPU is stored and transmitted in encrypted form. All data transmitted over the code or datapath is encrypted with an address-dependent non-linear encryption scheme. En- and decryptions are performed in the CPU core.
- **Address Scrambling:** Set of security mechanisms that ensure that physical addresses are scrambled before writing data to the memory.
- **Code- & Datapath Key Management:** Set of security mechanisms that ensure that keys used for the secure data path are derived correctly and securely. This includes address dependent key derivation functionality with an according strength of diffusion and confusion to achieve a good avalanche effect.

Note that the TOE does also support the Proximity Check feature against relay attacks on the TOE. The proximity check feature is an optional challenge response protocol on which the round trip time is measured by the terminal.

SF.LOG **Logical Protection**

SF.LOG implements measures to limit or eliminate the information that might be contained in the shape and amplitude of signals or in the time between events found by measuring such signals. This comprises the power consumption and signals on the other pads that are not intended by the terminal or the Security IC Embedded Software. Thereby SF.LOG prevents the disclosure of User Data or TSF data stored and/or processed in the security IC through the measurement of the power consumption or emanation and subsequent complex signal processing. The protection of the TOE comprises different features within the design that support the other portions of security functionality.

SF.COMP **Protection of Mode Control**

SF.COMP provides a control of the TOE modes. This includes the protection of electronic fuses stored in a protected memory area, and the possibility to store initialisation or pre-personalisation data in the so-called FabKey Area.

The control of the TOE modes prevent the abuse of test functions after TOE delivery. Additionally it also ensures that features used during the boot sequence to configure the TOE can not be abused. Hardware circuitry and the Boot Software determine whether the test functionality is available or not. If it is available, the TOE starts the IC Dedicated Test Software in the System Mode. Otherwise, the TOE switches to the User Mode or System Mode and starts execution of the MIFARE DESFire Software.

The switch to the IC Dedicated Test Software is prevented after TOE delivery because specific electronic fuses guarantee that the IC Dedicated Test Software cannot be selected. The System Mode is the more privileged TOE mode, the User Mode is the less privileged TOE mode. The System Mode HAL Software as part of the IC Dedicated Support Software is executed in System Mode. For the MIFARE DESFire Software, only the User Mode is available. The protection of the electronic fuses especially ensures that configuration options with regard to the security functionality cannot be changed, abused or influenced in any way in User Mode. SF.COMP ensures that activation or deactivation of security features cannot be influenced by the MIFARE DESFire Software.

SF.COMP limits the capabilities of the test functions and provides test personnel during phase 3 with the capability to store the identification and/or pre-personalization data in the EEPROM.

7.2 TOE Summary Specification Rationale

7.2.1 Rationale for assurance measures

The assurance measures defined in section 6.2 are considered to fulfil the assurance requirements of the Common Criteria, Part 3 [4] at level EAL5. Since the Protection Profile defines assurance measures that are suitable to fulfil the requirements of EAL4, all input deliverables as listed in section 6.2 shall be sufficient to fulfil

the assurance requirements of the Protection Profile. The assurance measures are defined especially for the development and production of Smartcard ICs and observe also the refinements made in the Protection Profile.

As already explained in the Protection Profile, annex 7.1, the development and production process of a smartcard IC is complex. Regarding the great number of assurance measures, a detailed mapping of the assurance measures to the assurance requirements is beyond the scope of this Security Target. Nevertheless the suitability of the assurance measures is subject of different evaluation tasks. The documents "Quality Management Manual" and "Security Management Manual" describe the general benchmark of NXP.

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 self-protection and non-bypassability of the TOE is implemented by internal integrity checks of the stored data e.g. SS.ACC_CTRL, appropriate configuration of the hardware platform by enabling countermeasures controlled by the software and by countermeasures implemented in the software. SS.TRANSACTION, SS.MAC, SS.TRANSACTION_MAC and SS.ENCRYPTION provide protection against logical interference based on the control of transaction sequences and the integrity control of exchanged messages.

SS.AUTH requires an authentication before specific operations are allowed. SS.AUTH authentication either uses 128-bit AES cryptographic algorithm; according to FIPS PUB 197 [6] or 3-key Triple-DES according to FIPS PUB 46-3 [7]. Furthermore 16 Byte random challenges are used for SS.AUTH. Any context change or error resets the authentication status to prevent interference between applications and the bypass of the authentication request. SS.ACC_CTRL is also implemented in a way that supports the protection against interference, logical tampering and bypass. SS.NO_TRACE contributes to the self-protection of the TOE by protecting card specific data. Using SS.NO_TRACE and SS.ACC_CTRL ensures that traceability of end-users is prevented.

8 Bibliography

[1] A proposal for: Functionality classes for random number generators, Version 2.0, 18. September 2011.

[2] Common Criteria for Information Technology Security Evaluation, Part 1 – Introduction and general model - Version 3.1 CCMB-2012-09-001, Revision 4, September 2012.

[3] Common Criteria for Information Technology Security Evaluation, Part 2 – Security functional components, Version 3.1 CCMB-2012-09-002, Revision 4, September 2012.

[4] Common Criteria for Information Technology Security Evaluation, Part 3 – Security assurance components, Version 3.1 CCMB-2012-09-003, Revision 4, September 2012.

[5] Common Methodology for Information Technology Security Evaluation, Evaluation Methodology, Version 3.1 CCMB-2012-09-004, Revision 4, September 2012.

[6] FIPS PUB 197 FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION, ADVANCED ENCRYPTION STANDARD (AES), National Institute of Standards and Technology, 2001 November 26.

[7] FIPS PUB 46-3 FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION DATA ENCRYPTION STANDARD (DES) Reaffirmed 1999 October 25.

[8] MF3Dx2 - Information on Guidance and Operation, Guidance and Operation Manual, NXP Semiconductors, Document number 274811.

[9] MF3Dx2 - MIFARE DESFire EV2 - Security Target, Evaluation Documentation, NXP Semiconductors.

[10] MF3Dx2 - MIFARE DESFire EV2 contactless multi-application IC, Product Data Sheet, NXP Semiconductors, Document number 226030.

[11] NIST Special Publication 800-38A Recommendation for BlockCipher Modes of Operation. http://csrc.nist.gov/publications/nistpubs/800-38a/sp800-38a.pdf.

[12] NIST Special Publication 800-38B Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication. http://csrc.nist.gov/publications/nistpubs/800-38B/SP_800-38B.pdf.

[13] Security IC Platform Protection Profile with Augmentation Packages, registered and certified by Bundesamt fuer Sicherheit in der Informationstechnik (BSI) under the reference BSI-CC-PP-0084-2014, Version 1.0, 13 January 2014.

[14] ISO/IEC 14443-1:2000 Identification cards – Contactless integrated circuit(s) cards – Proximity cards – Part 1: Physical characteristics, 2008.

[15] ISO/IEC 14443-4:2001 Identification cards – Contactless integrated circuit(s) cards – Proximity cards – Part 4: Transmission protocol, 2008.

[16] ISO/IEC 14443-2:2001 Identification cards – Contactless integrated circuit(s) cards – Proximity cards – Part 2: Radio frequency power and signal interface, 2010.

[17] ISO/IEC 14443-3:2001 Identification cards – Contactless integrated circuit(s) cards – Proximity cards – Part 3: Initialization and anticollision, 2011.

[18] Bundesamt für Sicherheit in der Informationstechnik. Anwendungshinweise und Interpretationen zum Schema, AIS20: Funktionalitätsklassen und Evaluationsmethodologie für deterministische Zufallszahlengeneratoren, Bundesamt für Sicherheit in der In formationstechnik. Version 2.0, December 2, 1999.

[19] Bundesamt für Sicherheit in der Informationstechnik. Anwendungshinweise und Interpretationen zum Schema, AIS31: Funktionalitätsklassen und Evaluationsmethodologie für physikalische Zufallszahlengeneratoren, Bundesamt für Sicherheit in der In formationstechnik. Version 2.0, September 18, 2011.

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.

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 accepts 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.

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.

Evaluation products – This product is provided on an "as is" and "with all faults" basis for evaluation purposes only. NXP Semiconductors, its affiliates and their suppliers expressly disclaim all warranties, whether express, implied or statutory, including but not limited to the implied warranties of non-infringement, merchantability and fitness for a particular purpose. The entire risk as to the quality, or arising out of the use or performance, of this product remains with customer.

In no event shall NXP Semiconductors, its affiliates or their suppliers be liable to customer for any special, indirect, consequential, punitive or incidental damages (including without limitation damages for loss of business, business interruption, loss of use, loss of data or information, and the like) arising out the use of or inability to use the product, whether or not based on tort (including negligence), strict liability, breach of contract, breach of warranty or any other theory, even if advised of the possibility of such damages.

Notwithstanding any damages that customer might incur for any reason whatsoever (including without limitation, all damages referenced above and all direct or general damages), the entire liability of NXP Semiconductors, its affiliates and their suppliers and customer's exclusive remedy for all of the foregoing shall be limited to actual damages incurred by customer based on reasonable reliance up to the greater of the amount actually paid by customer for the product or five dollars (US$5.00). The foregoing limitations, exclusions and disclaimers shall apply to the maximum extent permitted by applicable law, even if any remedy fails of its essential purpose.

9.3 Licenses

ICs with DPA Countermeasures functionality

NXP ICs containing functionality implementing countermeasures to Differential Power Analysis and Simple Power Analysis are produced and sold under applicable license from Cryptography Research, Inc.

9.4 Patents

Notice is herewith given that the subject device uses one or more of the following patents and that each of these patents may have corresponding patents in other jurisdictions.
<Patent ID> – owned by <Company name>

9.5 Trademarks

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

MIFARE – is a trademark of NXP B.V.

10 Contents

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

Date of release: 2016-04-29

Document identifier: