MF3D(H)x3

Security Target Lite

Rev. 2.0 — 5 June 2020 **Evaluation document**

NSCIB-CC-0011955 **PUBLIC**

Document information

Information	Content
Keywords	Common Criteria, Security Target, MF3D(H)x3, MIFARE DESFire EV3
Abstract	Evaluation of the MF3D(H)x3 developed and provided by NXP Semiconductors, Business Line Connectivity & Security, according to the Common Criteria for Information Technology Evaluation Version 3.1 at EAL5 augmented

Revision History

Revision	Date	Description
2.0	2020-06-05	Derived from full Security Target Rev. 2.1
1.0	2020-03-17	Derived from full Security Target Rev. 1.3

1 Introduction

1.1 ST Reference

MF3D(H)x3 Security Target Lite, Version 2.0, NXP Semiconductors, 5 June 2020.

1.2 TOE Reference

NXP Secure Smart Card Controller MF3D(H)x3.

1.3 TOE Overview

NXP has developed the MF3D(H)x3 to be used with Proximity Coupling Devices (PCDs, also called "terminal") according to ISO 14443 Type A. The communication protocol complies to part ISO 14443-4. The MF3D(H)x3 is primarily designed for secure contactless transport applications and related loyalty programs as well as access control management 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 Flash and ROM memory and provides an operating system with a set of functions, used to manage the various kinds of data files stored in Flash 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 CPU, 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 TOE 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.

1.3.1 Required non-TOE Hardware/Software/Firmware

The TOE requires an ISO 14443 card terminal to be provided with power and to receive adequate commands.

1.4 TOE Description

1.4.1 Physical Scope of the TOE

The Target of Evaluation (TOE) is the smartcard integrated circuit named MF3D(H)x3 in combination with a fixed software package, the IC Dedicated Software. The TOE includes IC manufacturer proprietary IC Dedicated Test Software and IC Dedicated

Support Software, according to the terminology used in the Security IC Protection Profile [6].

Table 1. TOE deliverables

Type	Name	Release	Form of delivery
IC Hardware	MF3D(H)x3 Hardware	A1.A04	Sawn wafer (FFC), modules (MOA4, MOA8, MOB10)
IC Dedicated Test Software	Test Software	A1.A04	On-chip software
IC Dedicated Support Software	Boot Software	A1.A04	On-chip software
	Firmware	A1.A04	On-chip software
	MIFARE DESFire Software	3.0	On-chip software
Document	MF3D(H)x3, MIFARE DESFire EV3 contactless smartcard IC, Product data sheet [7]	3.0	Electronic document (PDF via NXP DocStore)
Document	MF3D(H)x3C, MIFARE DESFire EV3C contactless smartcard IC, Product data sheet [8] (for MFC unlocked configurations only)	3.0	Electronic document (PDF via NXP DocStore)
Document	MF3D(H)x3, MIFARE DESFire EV3 Post Delivery Configuration, Preliminary data sheet addendum [9]	2.0	Electronic document (PDF via NXP DocStore)
Document	MF3D(H)x3, Wafer and Delivery Specification, Product data sheet addendum [10]	3.0	Electronic document (PDF via NXP DocStore)
Document	MF3D(H)x3, Information on Guidance and Operation, Guidance and Operation Manual [11]	1.0	Electronic document (PDF via NXP DocStore)

The TOE (hardware) is shipped to the customer by NXP. The available documentation can be downloaded by customers in PDF format directly from the NXP DocStore.

The customer can check the version of the IC Hardware and IC Dedicated Software by using the GetVersion APDU as described in Section 4.1 of the Wafer and Delivery Specification [10]. Additionaly, the originality of the TOE can be determined by authentication with an Originality Key, as described in Section 4.2 of the Wafer and Delivery Specification [10].

1.4.1.1 Evaluated Configurations

The TOE is available in various configurations. Each configuration has a different commerical type name. A commercial type name for the TOE has the following general format:

- MF3Dcxeywdpp(p)/svff

The following table illustrates the commerical type names that are subject of the evaluation:

Table 2. Variable definitions for commercial type names

Identifier	Description	Assignment	Meaning
c	input capacitance	<omitted> H	17 pF 70 pF
x	memory size	0 2 4 8	0.5 KB of non-volatile memory 2 KB of non-volatile memory 4 KB of non-volatile memory 8 KB of non-volatile memory
e	evolution	3	MIFARE DESFire EV3
y	MFC functionality	0 1 C D P Q	MIFARE Classic locked (7-byte UID) MIFARE Classic locked (10-byte UID) MIFARE Classic unlocked (7-byte UID) MIFARE Classic unlocked (10-byte UID) EV2-AES only (No MFC) (7-byte UID) EV2-AES only (No MFC) (10-byte UID)
w	wafer fab code	0 1	multiple (for modules) GlobalFoundries
d	fixed value	D	
pp(p)	package type	A4 A8 A10 UD UF	MOA4 module MOA8 module MOB10 module 120µm wafer 75µm wafer
/	separator		
s	SW minor version (higher nibble)	0	SW minor version information
v	SW minor version (lower nibble)	0	SW minor version information
ff	Type ID	A...Z, 0...9 <omitted>	customer data identification default type without customer data

All commercial type names are subject to this evaluation. However the identifier "MF3D(H)x3" will be used in the remainder of this document to make referencing easier. All information and security functionality described in this Security Target applies to all commercial types.

1.4.2 Logical Scope of the TOE

1.4.2.1 Hardware Description

The CPU of the MF3D(H)x3 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, contactless communication, Flash, timers, the DES co-processor and the AES co-processor. The communication with the MF3D(H)x3 can be performed through the contactless interface.

The AES co-processor supports AES operations with a key length of 128 bit. The Triple-DES co-processor supports Triple-DES operations with key lengths of 112 bits and 168 bits.

A hardware Random Number Generator provides true random numbers which are used to seed deterministic random number generators, used internally by the MIFARE DESFire functionality for security purposes.

1.4.2.2 Software Description

The IC Dedicated Test Software (Test ROM Software) located in 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 and shutdown functions to ensure that security relevant test operations cannot be executed illegally after phase 3 of the TOE Life cycle.

The TOE also contains IC Dedicated Support Software. The Boot Software which is stored in 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 MF3D(H)x3 is primarily designed for secure contactless 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.
- State-of-the-art Mutual Authentication and Secure Messaging as introduced in DESFire EV2.
- 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.
- Secure Dynamic Messaging feature which allows confidential and integrity protected data exchange without requiring a preceding authentication.
- MIFARE DESFire D40 backward compatible mode for authentication and secure messaging.

- MIFARE DESFire EV1 backward compatible mode for authentication and secure messaging.
- MIFARE Classic compatible mode which allows mapping of MIFARE Classic Blocks into the MIFARE DESFire file system.

The recommended authentication and secure messaging is called EV2 secure messaging, which is covered by the Security Functional Requirement mentioned in this Security Target. For DESFire EV1 backward compatible mode, the certification scope is limited to the AES mode (both for authentication and secure messaging) and 3TDEA (only authentication). Please note that the MIFARE DESFire D40 backward compatible mode is not part of any SFR and therefore not in the certification scope. Also the MIFARE Classic compatible mode, including all DESFire applications that enable MIFARE Classic mapping, are excluded from the certification scope.

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 privacy-related information to unauthorized users.

1.4.2.3 Documentation

The Product Data Sheet [7] 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 develop applications using the TOE.

The Product Data Sheet is supported by a User Guidance Manual [11] which gives additional guidance with regards to the secure usage of the TOE.

The Wafer and Deliver Specification data sheet addendum [10] gives additional information regarding the wafer dimensions, TOE identification and delivery processes.

1.4.3 Life Cycle and Delivery of the TOE

The life-cycle phases are organized according to the Security IC Platform Protection Profile with Augmentation Packages [6], 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 MF3D(H)x3 chip will be embedded in a credit card (meaning ID-1 sized) plastic card (micro-module embedded into the plastic card) or another supported 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 the Protection Profile [6], NXP will deliver the TOE at the end of Phase 6. Therefore the TOE evaluation perimeter comprising the development

and production environment of the TOE, consists of life-cycle phases 1 - 6. The TOE is a fully integrated composite product comprised of the underlying security IC hardware combined with the embedded software developed by NXP. Therefore, Phase 5 is fully under control of NXP and does not involve data exchange with other parties.

NXP also provides a commercial option to configure the TOE on behalf of the customer in order to personalize before the usage. Alternatively, the customer can also finalize the partially personalized TOE after delivery. In case that all required security anchors (key material) are already installed during personalization by NXP, the customer can finalize the personalization of the file system content relying on the operational security features of the TOE.

The TOE Software is embedded in the TOE during the TOE evaluation perimeter (life-cycle phases 1 - 6) and the TOE does not allow the modification of installation of any piece of IC Embedded Software after TOE delivery. Moreover, the TOE is being locked to the user operating mode before TOE delivery at the end of Phase 6.

The TOE is able to control two different logical phases. After production of the chip every start-up will lead to the initial operating mode. In the initial operating mode the production test shall be performed and the TOE is trimmed and initialized. The selection of the required variant is part of the initialization. At the end of the production test, the access to the test and initialization software is disabled. Subsequent start-ups of the chip will always enter the user operating mode with the CPU executing the TOE operating system software. The TOE will stay in the user operating mode until the end of its life-time.
In exceptional cases, which impact the integrity of the TOE in a non-recoverable way (typically if the TOE configuration is corrupted or TOE faces physical damage) the TOE switches into the mute or freeze operating mode. In those modes the TOE is effectively unusable.

1.4.4 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. The only exception is that customer specific keys can be installed using trust provisioning services in Phase 6. In this case the customer can finalize the personalization at the end of Phase 6, already relying on the TOE provided operational security services. 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 contactless smart cards according to ISO 14443. 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.

1.4.5 Interface of the TOE

The electrical interface of the TOE are the pads to connect the RF antenna, which allows communication according to ISO 14443 Type A. The communication protocol complies to part ISO 14443-4. The functional interface is defined by the commands implemented by the TOE and described in the product data sheet.

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, Revision 5, CCMB-2017-04-001, April 2017 [2].
- Common Criteria for Information Technology Security Evaluation, Part 2: Security functional components, Version 3.1, Revision 5, CCMB-2017-04-002, April 2017 [3].
- Common Criteria for Information Technology Security Evaluation, Part 3: Security assurance components, Version 3.1, Revision 5, CCMB-2017-04-003, April 2017 [4].

For the evaluation the following methodology will be used:

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

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

2.2 PP Claim

This Security Target claims strict conformance to the following Protection Profile:

- Security IC Platform Protection Profile with Augmentation Packages, Registered and Certified by Bundesamt für Sicherheit in der Informationstechnik (BSI) under the reference BSI-CC-PP-0084-2014, Version 1.0, 13 January 2014 [6].

2.3 Package Claim

This Security Target claims conformance to the assurance package EAL5 augmented with AVA_VAN.5 and ALC_DVS.2.

2.4 Conformance Claim Rationale

As the Protection Profile [6] requires strict conformance, no conformance claim requirement is needed in this Security Target.

3 Security Problem Definition

This section lists the assets, threats, organisational security policies and assumptions from the Protection Profile [6] and describes extensions to these elements in detail.

3.1 Description of Assets

The assets to be protected (related to standard functionality) are described in Section 3.1 of the Protection Profile [6] and are listed below:

- The user data of the Composite TOE.
- The Security IC Embedded Software, stored and in operation.
- The security services provided by the TOE for the Security IC Embedded Software.

These assets are related to the following high-level security concerns:

- Integrity of user data of the Composite TOE.
- Confidentiality of user data of the Composite TOE being stored in the TOE's protected memory areas.
- 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 these assets the TOE shall self-protect its security functionality. Critical information about the security functionality shall be protected by the development environment and the operational environment. Critical information may include:

- Logical design data, physical design data, IC Dedicated Software, and configuration data.
- Initialisation Data and Pre-personalisation Data, specific development aids, test and characterisation related data, material for software development support, and photomasks.

For details see Section 3.1 of the Protection Profile [6].

3.2 Threats

All threats for the TOE which are defined in section 3.2 of the Protection Profile are applied to this Security Target and are listed in Table 3.

Table 3. Threats defined in the Protection Profile (PP-0084)

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

For details see Section 3.2 of the Protection Profile [6].

The following additional threats are defined in this Security Target:

Table 4. Additional threats defined in this Security Target

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

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 Organisational Security Policies

All organisational security policies defined in the Protection Profile are valid for this Security Target and are listed in Table 5. For details see Section 3.3 of the Protection Profile [6].

Table 5. Organisational security policies defined in the Protection Profile (PP-0084)

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

This Security Target defines additional organisational 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 'Untraceability of end-users' as specified below.

Table 6. Additional organisational security policies defined in this Security Target

Name	Title
P.Encryption	Confidentiality during communication
P.MAC	Integrity during communication
P.Transaction	Transaction mechanism

Name	Title
P.No-Trace	Untraceability of end-users

P.Encryption **Confidentiality during communication**

The TOE shall provide the possibility to protect selected data elements from eavesdropping during contactless communication.

P.MAC **Integrity during communication**

The TOE shall provide the possibility to protect the contactless 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 **Untraceability 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 contactless 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 the Protection Profile are valid for this Security Target and are listed in Table 7. For details see Section 3.4 of the Protection Profile [6].

Table 7. Assumptions defined in the Protection Profile (PP-0084)

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

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

Table 8. Additional assumptions defined in this Security Target

Name	Title
A.Secure-Values	Usage of secure values
A.Terminal-Support	Terminal Support

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** 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 AIS20/31 [1] for the authentication.

The additional assumptions as defined above are required for the correct functioning of the DESFire security functionality. As the Protection Profile [6] does not cover this kind of functionality, the additional assumptions neither mitigate a threat (or a part of a threat) meant to be addressed by security objectives for the TOE in the Protection Profile [6], nor fulfil an OSP (or part of an OSP) meant to be addressed by security objectives for the TOE in the Protection Profile [6].

4 Security Objectives

4.1 Security Objectives for the TOE

All security ojectives for the TOE which are defined in section 4.1 of the Protection Profile are applied to this Security Target and are listed in Table 9.

Table 9. Security Objectives of the TOE (PP-0084)

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

Regarding the Application Notes 8 and 9 in the Protection Profile [6], additional security objectives that are based on additional functionality provided by the TOE are defined below:

Table 10. Additional security objectives defined in this Security Target

Name	Title
O.Access-Control	Access Control
O.Authentication	Authentication
O.Encryption	Confidential Communication
O.MAC	Integrity-Protected Communication
O.No-Trace	Preventing Traceability
O.Transaction	Transaction Mechanism
O.Type-Consistency	Data Type Consistency

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.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 privacy-related information that is suitable for tracing an end-user by an unauthorised subject.
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.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.

4.2 Security Objectives for the Security IC Embedded Software

All security objectives for the Security IC Embedded Software which are defined in section 4.2 of the Protection Profile are applied to this Security Target and are listed in Table 11.

Table 11. Security Objectives for the Security IC Embedded Software (PP-0084)

Name	Title
OE.Resp-Appl	Treatment of User Data

4.3 Security Objectives for the Operational Environment

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 are listed in Table 12.

Table 12. Security Objectives for the Operational Environment (PP-0084)

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

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

Table 13. Additional security objectives for the operational environment defined in this Security Target

Name	Title
OE.Secure-Values	Generation of secure values
OE.Terminal-Support	Terminal support to ensure integrity, confidentiality and use of random numbers

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 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 AIS20/31 [1] for the authentication.

The additional security objectives for the operational environment as defined above are required for the correct functioning of the MIFARE DESFire security functionality. As the Protection Profile [6] does not cover this kind of functionality, the additional objectives neither mitigate a threat (or a part of a threat) meant to be addressed by security objectives for the TOE in the Protection Profile [6], nor fulfil an OSP (or part of an OSP) meant to be addressed by security objectives for the TOE in the Protection Profile [6].

4.4 Security Objectives Rationale

Section 4.4 in the Protection Profile [6] provides a rationale how the threats, organisational security policies and assumptions are addressed by the security objectives defined in the Protection Profile. This rationale is not repeated here.

The following table 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 [6].

Table 14. Security Problem Definition mapping to Security Objective

Security Problem Definition	Security Objective
T.Data-Modification	O.Access-Control O.Type-Consistency OE.Terminal-Support
T.Impersonate	O.Authentication
T.Cloning	O.Access-Control O.Authentication
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

The rationale for the mapping is given below:

Justification related to T.Data-Modification:

Security 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:

Security 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:

Security 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:

Security 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:

Security 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:

Security 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:

Security 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:

Security 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:

Security 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 policies and the additional assumptions show that they do not contradict the rationale already given in the Protection Profile [6] for the assumptions, policy and threats defined there.

5 Extended Components Definition

To define the Secure Dynamic Messaging functionality of the TOE, an additional component FDP_ETC.3 of the family FDP_ETC (export from the TOE) of the class FDP (user data protection) is defined.

As defined in CC Part 2 [3], the FDP class addresses user data protection. The FDP_ETC family defines functions for TSF-mediated exporting of user data from the TOE such that its security attributes and protection either can be explicitly preserved or can be ignored once it has been exported. The extended component FDP_ETC.3 (Export of user data in unauthenticated state) addresses a similar concern but does not require a TOE enforcement of an access control SFP(s) and/or information flow control SFP(s) as the already defined components of the FDP_ETC family.

Note that the Protection Profile [6] 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.

5.1 Export of user data in unauthenticated state (FDP_ETC.3)

The class and family behaviour of FDP_ETC are already defined in CC Part 2 [3].

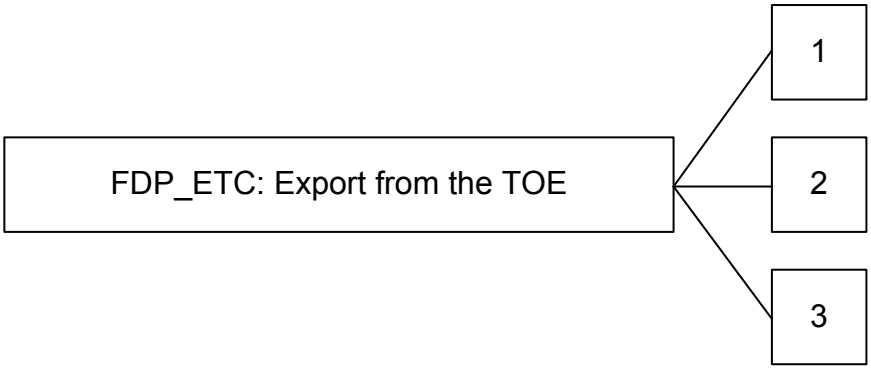

Figure 1. Component levelling of Extended Component FDP_ETC

FDP_ETC	Export from the TOE
Management:	FDP_ETC.3 There are no management activities foreseen.
Audit:	FDP_ETC.3 There are no actions defined to be auditable.
FDP_ETC.3	**Export of user data in unauthenticated state**
Hierarchical to:	No other components.

Dependencies:	No dependencies.
FDP_ETC.3.1	**The TSF shall export the following pieces of user data: [assignment: *pieces of user data*] with the following user data's associated security attributes: [assignment: *list of security attributes*].**
FDP_ETC.3.2	**The TSF shall ensure that the security attributes, when exported outside the TOE, are unambiguously associated with the exported user data.**
FDP_ETC.3.3	**The TSF shall enforce the following rules when user data is exported from the TOE: [assignment: *additional exportation control rules*]**

The extended component is defined to capture the Secure Dynamic Messaging feature provided by the TOE, which allows for the encrypted and authenticated extraction of user data without the need of establishing a trusted channel beforehand. Due to this specific property, the existing data export SFRs FDP_ETC.1 and FDP_ETC.2 did not apply well.

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 [6] and in this Security Target, respectively.

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

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. 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

6.1.1 Security Functional Requirements from the Protection Profile

6.1.1.1 FAU_SAS.1

The TOE shall meet the requirement "Audit storage" as defined in the PP [6], and as specified below.

FAU_SAS.1	**Audit storage**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FAU_SAS.1.1	The TSF shall provide *the test process before TOE Delivery* with the capability to store *the Initialisation Data, Pre-personalisation Data, Customer-specific Data*[1] in the *non-volatile memory*[2].

1 [selection: *the Initialisation Data, Pre-personalisation Data, [assignment: other data]*]
2 [assignment: *type of persistent memory*]

6.1.1.2 FCS_RNG.1/PTG2

The TOE shall meet the requirement "Random number generation (Class PTG.2)" as defined in the PP [6] according to [1] , and as specified below.

FCS_RNG.1/PTG2	**Random number generation (Class PTG.2)**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FCS_RNG.1.1/PTG2	The TSF shall provide a *physical*[3] random number generator that implements:[4] (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*[5]. (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*[6]. 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/PTG2	The TSF shall provide *octets of bits*[7] that meet:

3 [selection: *physical, hybrid physical, hybrid deterministic*]
4 [assignment: *list of security capabilities*]
5 [selection: *prevents the output of any internal random number that depends on some raw random numbers that have been generated after the total failure of the entropy source, generates the internal random numbers with a post-processing algorithm of class DRG.2 as long as its internal state entropy guarantees the claimed output entropy*]
6 [selection: *externally, at regular intervals, continuously, applied upon specified internal events*]
7 [selection: *bits, octets of bits, numbers [assignment: format of the numbers]*]

(PTG.2.6) Test procedure A[8] does not distinguish the internal random numbers from output sequences of an ideal RNG.

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

6.1.1.3 FCS_RNG.1/DRG3

The TOE shall meet the requirement "Random number generation (Class DRG.3)" as defined below according to [1].

FCS_RNG.1/DRG3	**Random number generation (Class DRG.3)**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FCS_RNG.1.1/ DRG3	The TSF shall provide a *deterministic*[9] random number generator that implements:[10]
	(DRG.3.1) If initialized with a random seed *using a PTRNG of class PTG.2 as random source*[11], the internal state of the RNG shall *have at least 256 bit of entropy*[12].
	(DRG.3.2) The RNG provides forward secrecy.
	(DRG.3.3) The RNG provides backward secrecy even if the current internal state is known.
FCS_RNG.1.2/ DRG3	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*[13], generates output for which[14] 2^{48} strings of bit length 128 are mutually different with probability[15] *at least* $1 - 2^{-24}$.
	(DRG.3.5) Statistical test suites cannot practically distinguish the random numbers from output sequences of an ideal RNG. The

8 [assignment: *additional standard test suites*]. Assignment is empty as per Application Note 44 of the PP.
9 [selection: *physical, non-physical true, deterministic, hybrid physical, hybrid deterministic*]
10 [assignment: *list of security capabilities*]
11 [selection: *using a PTRNG of class PTG.2 as random source, using a PTRNG of class PTG.3 as random source, using an NPTRNG of class NTG.1 [assignment: other requirements for seeding]*]
12 [selection: *have [assignment: amount of entropy], have [assignment: work factor], require [assignment: guess work]*]
13 [assignment: *requirements for seeding*]
14 [assignment: *number of strings*]
15 [assignment: *probability*]

random numbers must pass test procedure A *and no additional test suites*[16].

6.1.1.4 FDP_SDC.1

The TOE shall meet the requirement "Stored data confidentiality" as defined in the PP [6], and as specified below.

FDP_SDC.1	**Stored data confidentiality**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FDP_SDC.1.1	The TSF shall ensure the confidentiality of the information of the user data while it is stored in the *RAM and non-volatile memory*[17].

6.1.1.5 FDP_SDI.2

The TOE shall meet the requirement "Stored data integrity monitoring and action" as defined in the PP [6], and as specified below.

FDP_SDI.2	**Stored data integrity monitoring and action**
Hierarchical to:	FDP_SDI.1 Stored data integrity monitoring
Dependencies:	No dependencies.
FDP_SDI.2.1	The TSF shall monitor user data stored in containers controlled by the TSF for *modification, deletion, repetition or loss of data*[18] on all objects, based on the following attributes: *integrity check information associated with the data storied in memories*[19].
FDP_SDI.2.2	Upon detection of a data integrity error, the TSF shall *perform an error correction if possible or trigger a Security Reset if not*[20].

6.1.2 Security Functional Requirements regarding Access Control

6.1.2.1 DESFire Access Control Policy

The Security Function Policy (SFP) *DESFire Access Control Policy* uses the definitions listed in this paragraph. The defined subjects are:

16 [assignment: additional test suites]
17 [assignment: *memory area*]
18 [assignment: *integrity errors*]
19 [assignment: *user data attributes*]
20 [assignment: *action to be taken*]

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 the AppDAMDefaultKey, 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 multiple DelApplications 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 access 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 defined for the *DESFire Access Control Policy* 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 Key Set 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.

6.1.2.2 FDP_ACC.1/DF

The TOE shall meet the requirement "Subset access control" 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*[21] on *all subjects, objects, operations and attributes defined by the DESFire Access Control Policy*[22].

6.1.2.3 FDP_ACF.1/DF

The TOE shall meet the requirement "Security attribute based access control" 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 initialisation
FDP_ACF.1.1/DF	The TSF shall enforce the *DESFire Access Control Policy*[23] to objects based on the following: *all subjects, objects and attributes*[24].
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:[25] 1. *The Admin is allowed to perform Application.Create and Application.Delete.* 2. *The Admin is allowed to perform DelApplication.Delete.* 3. *The AppMgr is allowed to perform File.Create and File.Delete.*

21 [assignment: *access control SFP*]
22 [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]
23 [assignment: *access control SFP*]
24 [assignment: *list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes*]
25 [assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]

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

FDP_ACF.1.3/DF — The TSF shall explicitly authorise access of subjects to objects based on the following additional rules:[26]

1. *The AppMgr is allowed to Application.Delete if the attribute PICCLevelData.PICCKeySettings grant this right.*
2. *The AppUser is allowed to perform File.Read or File.Write or File.ReadWrite or File.Change on File if the File.AccessRights grant these rights.*
3. *The Anybody is allowed to perform Application.Create if the PICCLevelData.PICCKeySettings grant this right.*
4. *The Anybody is allowed to perform File.Create and File.Delete if the Application.AppKeySettings grant these rights.*
5. *The 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:[27]

1. *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.*
2. *OrigKeyUser is not allowed to perform any operation on objects.*
3. *No one but Nobody is allowed to perform any operation on OriginalityKey.*

6.1.2.4 FDP_ITC.2/DF

The TOE shall meet the requirement "Import of user data with security attributes" 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*[28] when importing user data, controlled under the SFP, from outside of the TOE.

26 [assignment: *rules, based on security attributes, that explicitly authorise access of subjects to objects*]
27 [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*]
28 [assignment: *access control SFP(s) and/or information flow control SFP(s)*]

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*[29].

6.1.2.5 FMT_MSA.1/DF

The TOE shall meet the requirement "Management of security attributes" 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*[30] to restrict the ability to *modify or freeze and change*[31] the security attributes *of the objects PICCLevelData, Application and the security attribute File.AccessRights*[32] to *the Admin, AppMgr and AppChangeUser respectively*[33].
Refinement:	The detailed management abilities are: 1. *Only the Admin is allowed to perform PICCLevelData.Modify or PICCLevelData.Freeze on PICCLevelData.PICCKeySettings.* 2. *Only the AppMgr is allowed to perform Application.Modify or Application.Freeze on Application.AppKeySettings.* 3. *The AppChangeUser is allowed to perform File.Freeze on File.AccessRights.*

29 [assignment: *additional importation control rules*]
30 [assignment: *access control SFP(s), information flow control SFP(s)*]
31 [selection: *change_default, query, modify, delete, [assignment: other operations]*]
32 [assignment: *list of security attributes*]
33 [assignment: *the authorised identified roles*]

6.1.2.6 FMT_MSA.3/DF

The TOE shall meet the requirement "Static attribute initialization" 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*[34] to provide *permissive*[35] default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2/DF	The TSF shall allow the *no one but Nobody*[36] 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.

6.1.2.7 FMT_MTD.1/DF

The TOE shall meet the requirement "Management of TSF data" 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*[37] the *operations PICCMasterKey.Change, PICCMasterKey.Freeze, PICCAppDefaultKey.Change, AppMasterKey.Change, AppMasterKey.Freeze, AppChangeKey.Change*[38] to *the Admin, AppMgr and AppUser*[39].
Refinement:	The detailed management abilities are: 1. *Only the Admin is allowed to perform PICCMasterKey.Change or PICCMasterKey.Freeze.* 2. *The Admin is allowed to perform PICCAppDefaultKey.Change.* 3. *The Admin is allowed to perform PICCDAMAuthKey.Change.*

34 [assignment: *access control SFP, information flow control SFP*]
35 [selection, choose one of: *restrictive, permissive, [assignment: other property]*]
36 [assignment: *the authorised identified roles*]
37 [selection: *change_default, query, modify, delete, clear, [assignment: other operations]*]
38 [assignment: *list of TSF data*]
39 [assignment: *the authorised identified roles*]

4. *The Admin is allowed to perform PICCDAMENCKey.Change.*
5. *The Admin is allowed to perform PICCDAMMACKey.Change.*
6. *The AppMgr is allowed to perform AppMasterKey.Change and AppMasterKey.Freeze.*
7. *The AppMgr is allowed to perform AppChangeKey.Change.*
8. *The AppMgr is allowed to perform AppKey.Change.*
9. *The AppMgr is allowed to perform AppRollKey.Change.*
10. *The AppMgr is allowed to perform AppTransactionMACKey.Create and AppTransactionMACKey.Delete.*
11. *The AppChangeUser is allowed to perform AppChangeKey.Change.*
12. *The AppChangeUser is allowed to perform AppKey.Change.*
13. *The AppUser is allowed to perform AppKey.Change on AppKey if Application.AppKeySettings grant this right.*
14. *The AppUser is allowed to perform AppTransactionMACKey.Create and AppTransactionMACKey.Delete on AppTransactionMACKey if Application.AppKeySettings grant this right.*
15. *The AppRollUser is allowed to perform KeySet.Roll.*

6.1.2.8 FMT_SMF.1/DF

The TOE shall meet the requirement "Specification of Management Functions" 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 management functions:[40] • 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, Occurrence 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 Key Set • Creating or deleting an application, a delegated application or a file • Selection of the Virtual Card

40 [assignment: *list of management functions to be provided by the TSF*]

6.1.2.9 FMT_SMR.1/DF

The TOE shall meet the requirement "Security roles" 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*[41].
FMT_SMR.1.2/DF	The TSF shall be able to associate users with roles.

6.1.2.10 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 Security Functional Requirements regarding Confidentialty, Authentication and Integrity

6.1.3.1 FCS_COP.1/DF-DES

The TOE shall meet the requirement "Cryptographic Operation (DES)" as specified below.

FCS_COP.1/DF-DES	**Cryptographic Operation (DES)**
Hierarchical to:	No other components.

41 [assignment: the authorised identified roles]

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*[42] in accordance with the specified cryptographic algorithm *Triple-DES in one of the following modes of operation: CBC and 3-key Triple-DES*[43] and cryptographic key sizes *168 bit*[44] that meet the following:[45] • *NIST SP 800-67 [15] (TDES)* • *NIST SP 800-38A [13] (CBC mode)*

6.1.3.2 FCS_COP.1/DF-AES

The TOE shall meet the requirement "Cryptographic Operation (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*[46] in accordance with the specified cryptographic algorithm *Advanced Encryption Standard AES in one of the following modes of operation: CBC, CMAC*[47] and cryptographic key sizes *128 bits*[48] that meet the following:[49] • *FIPS PUB 197 [12] (AES)* • *NIST SP 800-38A [13] (CBC mode)* • *NIST SP 800-38B [14] (CMAC mode)*
Refinement:	For the MIFARE DESFire EV1 secure messaging the TOE uses the cryptographic algorithm for CMAC according to NIST Special Publication 800-38B [14] (CMAC mode) with the following modification: The TOE does not use the standard zero byte

42 [assignment: *list of cryptographic operations*]
43 [assignment: *cryptographic algorithm*]
44 [assignment: *cryptographic key sizes*]
45 [assignment: *list of standards*]
46 [assignment: *list of cryptographic operations*]
47 [assignment: *cryptographic algorithm*]
48 [assignment: *cryptographic key sizes*]
49 [assignment: *list of standards*]

IV instead it uses an IV defined by the previous cryptographic operation (chaining mode).

6.1.3.3 FCS_CKM.1/DF

The TOE shall meet the requirement "Cryptographic key generation" as specified below.

FCS_CKM.1/DF	**Cryptographic key generation**
Hierarchical to:	No other components.
Dependencies:	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction
FCS_CKM.1.1/DF	The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *EV1 Session Key Generation (for AES) and EV2 Session Key Generation*[50] and specified cryptographic key sizes *128 bit*[51] that meets the following: *MF3D(H)x3 datasheet* [7], *Section 6.3.7.7 (EV2) and 6.3.9.5 (EV1)*[52].

6.1.3.4 FCS_CKM.4/DF

The TOE shall meet the requirement "Cryptographic key destruction" 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*[53] that meets the following: *none*[54].

6.1.3.5 FIA_UAU.2/DF

The TOE shall meet the requirement "User authentication before any action" as specified below.

FIA_UAU.2/DF	**User authentication before any action**

50 [assignment: *cryptographic key generation algorithm*]
51 [assignment: *cryptographic key sizes*]
52 [assignment: *list of standards*]
53 [assignment: *cryptographic key destruction method*]
54 [assignment: *list of standards*]

Hierarchical to:	FIA_UAU.1 Timing of authentication
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.

6.1.3.6 FIA_UAU.3/DF

The TOE shall meet the requirement "Unforgeable authentication" as specified below.

FIA_UAU.3/DF	**Unforgeable authentication**
Hierarchical to:	No other components
Dependencies:	No dependencies
FIA_UAU.3.1/DF	The TSF shall *detect and prevent*[55] use of authentication data that has been forged by any user of the TSF.
FIA_UAU.3.2/DF	The TSF shall *detect and prevent*[56] use of authentication data that has been copied from any other user of the TSF.

6.1.3.7 FIA_UAU.5/DF

The TOE shall meet the requirement "Multiple authentication mechanisms" 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*[57] to support user authentication.
FIA_UAU.5.2/DF	The TSF shall authenticate any user's claimed identity according to the *following rules:*[58] • *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.*

55 [selection: *detect, prevent*]
56 [selection: *detect, prevent*]
57 [assignment: *list of multiple authentication mechanisms*]
58 [assignment: *rules describing how the multiple authentication mechanisms provide authentication*]

- *The cryptographic authentication is used to authorise the Administrator, Application Manager, Delegated Application Manager and Application User.*

6.1.3.8 FIA_UID.2/DF

The TOE shall meet the requirement "User identification before any action" as specified below.

FIA_UID.2/DF	**User identification before any action**
Hierarchical to:	FIA_UID.1 Timing of identification
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".

6.1.3.9 FPT_TDC.1/DF

The TOE shall meet the requirement "Inter-TSF basic TSF data consistency" 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*[59] when shared between the TSF and another trusted IT product.
FPT_TDC.1.2/DF	The TSF shall use *the rule: data files or values can only be modified by their dedicated type-specific operations honouring the type-specific boundaries*[60] when interpreting the TSF data from another trusted IT product.

59 [assignment: *list of TSF data types*]
60 [assignment: *list of interpretation rules to be applied by the TSF*]

6.1.3.10 FTP_TRP.1/DF

The TOE shall meet the requirement "Trusted path" 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*[61] 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, disclosure, or only modification*[62].
FTP_TRP.1.2/DF	The TSF shall permit *remote users*[63] 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*[64].

6.1.4 Security Functional Requirements regarding Robustness

6.1.4.1 FDP_ROL.1/DF

The TOE shall meet the requirement "Basic rollback" 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 *DESFire Access Control Policy*[65] to permit the rollback of the *operations that modify the value or data file objects*[66] on the *backup files*[67].
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*

61 [selection: *remote, local*]
62 [selection: *modification, disclosure, [assignment: other types of integrity or confidentiality violation]*]
63 [selection: *the TSF, local users, remote users*]
64 [selection: *initial user authentication, [assignment: other services for which trusted path is required]*]
65 [assignment: *access control SFP(s) and/or information flow control SFP(s)*]
66 [assignment: *list of operations*]
67 [assignment: *information and/or list of objects*]

limitative events: chip reset, select command, deselect command, explicit commit, explicit abort, command failure[68].

6.1.4.2 FPR_UNL.1/DF

The TOE shall meet the requirement "Unlinkability" 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*[69] are unable to determine whether *any operation of the TOE*[70] *were caused by the same user*[71].

6.1.4.3 FPT_RPL.1/DF

The TOE shall meet the requirement "Replay detection" 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*[72].
FPT_RPL.1.2/DF	The TSF shall perform *rejection of the request*[73] when replay is detected.

6.1.5 Security Functional Requirements regarding Secure Dynamic Messaging

6.1.5.1 FDP_ETC.3/DF

The TOE shall meet the requirement "Export of user data in unauthenticated state" as specified below.

FDP_ETC.3/DF	**Export of user data in unauthenticated state**

68 [assignment: *boundary limit to which rollback may be performed*]
69 [assignment: *set of users and/or subjects*]
70 [assignment: *list of operations*]
71 [selection: *were caused by the same user, are related as follows[assignment: list of relations]*]
72 [assignment: *list of identified entities*]
73 [assignment: *list of specific actions*]

Hierarchical to:	No other components.
Dependencies:	No dependencies.
FDP_ETC.3.1/DF	The TSF shall export the following pieces of user data: *a configurable subset of file data*[74] with the following user data's associated security attributes: *confidentiality, authenticity and replay protection for the configurable subset of the file data*[75].
FDP_ETC.3.2/DF	The TSF shall ensure that the security attributes, when exported outside the TOE, are unambiguously associated with the exported user data.
FDP_ETC.3.3/DF	The TSF shall enforce the following rules when user data is exported from the TOE: *plain export of file data in case that SDM is not activated for the file*[76].

6.2 Security Assurance Requirements

The following table lists all security assurance components that are valid for this Security Target.

Table 15. Security Assurance Requirements

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_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

74 [assignment: *pieces of user data*]
75 [assignment: *list of security attributes*]
76 [assignment: *additional exportation control rules*]

Name	Title
ASE_ECD.1	Extended components definition
ASE_REQ.2	Derived security requirements
ASE_TSS.1	TOE summary specification (TSS)
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

6.2.1 Refinements of the TOE Security Assurance Requirements

In compliance to Application Note 23 in the PP, this Security Target has to conform to all refinements of the security assurance requirements in the PP. Because the refinements in the PP 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.

Most of the security assurance components mentioned in the PP and in this Security Target have the same component level and therefore for these components the refinements from the PP are valid for this Security Target without change. The following two subsections apply the refinements to ALC_CMS.5 and ADV_FSP.5, which are different between the PP and this Security Target.

6.2.1.1 Refinements Regarding 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 original component ALC_CMS.4 can be found in section 6.2.1.3 of the Protection Profile and is not repeated 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. This rationale is not repeated here.

This Security Target defines additional SFRs for the TOE. 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.

Table 16. Security Functional Requirements mapping to Security Objectives

Name	Title
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 FMT_MTD.1/DF FMT_SMF.1/DF FMT_SMR.1/DF
O.Authentication	FCS_COP.1/DF-DES FCS_COP.1/DF-AES FCS_CKM.1/DF FIA_UID.2/DF FIA_UAU.2/DF FIA_UAU.3/DF FIA_UAU.5/DF FMT_SMF.1/DF FPT_RPL.1/DF FTP_TRP.1/DF
O.Encryption	FCS_CKM.1/DF FCS_CKM.4/DF FCS_COP.1/DF-AES FTP_TRP.1/DF FDP_ETC.3/DF
O.MAC	FCS_CKM.1/DF FCS_CKM.4/DF FCS_COP.1/DF-AES FPT_RPL.1/DF FTP_TRP.1/DF FDP_ETC.3/DF

Name	Title
O.Type-Consistency	FPT_TDC.1/DF
O.Transaction	FDP_ROL.1/DF
O.No-Trace	FPR_UNL.1/DF

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 SFRs 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 FCS_CKM.1/DF generates the session key used after 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 SFR FIA_UAU.3/DF prevents that forged authentication data can be used. 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". The SFR FCS_CKM.1/DF generates the session key used for encryption. FCS_CKM.4/DF requires that cryptographic keys used for encryption have to be removed after usage.

The TOE also provides Secure Dynamic Messaging service which allows encrypted and MACed data read without being in the authenticated state. FDP_ETC.3/DF requires user data export in unauthenticated state, and hence models the requirements to reach 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". The SFR

FCS_CKM.1/DF generates the session key used for the calculation. FCS_CKM.4/DF requires that cryptographic keys used for MAC operations have to be removed after usage. FPT_RPL.1/DF requires a replay detection for these data transfers.

The TOE also provides Secure Dynamic Messaging service which allows encrypted and MACed data read without being in the authenticated state. FDP_ETC.3/DF requires user data export in unauthenticated state, and hence models the requirements to reach 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 and their fullfilment are listed in the tables below:

Table 17. Dependencies of Security Functional Requirements (PP-0084)

SFR	Dependency	Fullfilled in ST
FAU_SAS.1	No dependencies.	No dependency
FCS_RNG.1/PTG2	No dependencies.	No dependency
FCS_RNG.1/DRG3	No dependencies.	No dependency
FDP_ITT.1	[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	No dependencies.	No dependency
FDP_SDI.2	No dependencies.	No dependency
FMT_LIM.1	FMT_LIM.2 Limited availability.	Yes
FMT_LIM.2	FMT_LIM.1 Limited capabilities.	Yes
FPT_FLS.1	No dependencies.	No dependency
FPT_ITT.1	No dependencies.	No dependency

SFR	Dependency	Fullfilled in ST
FPT_PHP.3	No dependencies.	No dependency
FRU_FLT.2	FPT_FLS.1 Failure with preservation of secure state.	Yes

Table 18. Dependencies of Security Functional Requirements (Security Target)

SFR	Dependency	Fullfilled in ST
FCS_CKM.1/DF	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction	Yes, by FCS_COP.1/DF-DES, FCS_COP.1/DF-AES, FCS_CKM.4/DF.
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_CKM.1/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, FCS_CKM.1/DF, FCS_CKM.4/DF.
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, FCS_CKM.1/DF, 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 initialisation	Yes, by FDP_ACC.1/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, FTP_TRP.1/DF, 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
FDP_ETC.3/DF	No dependencies.	No dependency
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.3/DF	No dependencies	No dependency
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, FMT_SMR.1/DF, 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, FMT_SMR.1/DF.

SFR	Dependency	Fullfilled in ST
FMT_MTD.1/DF	FMT_SMR.1 Security roles, FMT_SMF.1 Specification of Management Functions	Yes, by FMT_SMR.1/ DF, FMT_SMF.1/DF.
FMT_SMF.1/DF	No dependencies.	No dependency
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

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. 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 "low-level 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 TOE Security Functionality (TSF) directly corresponds to the TOE security functional requirements defined in Section 6. The table below lists the TSF of the TOE.

Table 19. Portions of the TSF

TSF portion	Title	Description
TSF.Service	Service functionality not related to DESFire functionality	This portion of the TSF comprises services like random number generation and provides mechanisms to store initialization, pre-personalization, and/or other data on the TOE.
TSF.Protection	General security measures to protect the TSF	This portion of the TSF comprises physical and logical protection to avoid information leakage and detect fault injection. It defines resets in case an error or attack was detected.
TSF.Control	Operating conditions, memory and hardware access control	This portion of the TSF controls the operating conditions.
TSF.DESFire	DESFire functionality	This portion of the TSF comprises all DESFire related security functionality such as cryptographic algorithms used by DESFire,impements the DESFire Access Control Policy and security functionality related to DESFire confidentiality, authentication, integrity and robustness.

The TSF are described in more detail in the following sections and the relation to the security functional requirements is shown.

7.2 TOE Summary Specification Rationale

7.2.1 Mapping of Security Functional Requirements and TOE Security Functionality

SFR	TSF.Service	TSF.Protection	TSF.Control	TSF.DESFire	Description
Security Functional Requirements from the Protection Profile					
FRU_FLT.2			X		Limited fault tolerance
FPT_FLS.1			X		Failure with preservation of secure state
FMT_LIM.1			X		Limited capabilities

SFR	TSF.Service	TSF.Protection	TSF.Control	TSF.DESFire	Description
FMT_LIM.2			X		Limited availability
FAU_SAS.1	X				Audit storage
FDP_SDC.1		X			Stored data confidentiality
FDP_SDI.2		X			Stored data integrity monitoring and action
FPT_PHP.3		X			Resistance to physical attack
FDP_ITT.1		X			Basic internal transfer protection
FPT_ITT.1		X			Basic internal TSF data transfer protection
FDP_IFC.1		X			Subset information flow control
FCS_RNG.1/PTG2	X				Random number generation (Class PTG.2)
FCS_RNG.1/DRG3	X				Random number generation (Class DRG.3)
Security Functional Requirements regarding Access Control					
FDP_ACC.1/DF				X	Subset access control
FDP_ACF.1/DF				X	Security attribute based access control
FDP_ITC.2/DF				X	Import of user data with security attributes
FMT_MSA.1/DF				X	Management of security attributes
FMT_MSA.3/DF				X	Static attribute initialization
FMT_MTD.1/DF				X	Management of TSF data
FMT_SMF.1/DF				X	Specification of Management Functions
FMT_SMR.1/DF				X	Security roles
Security Functional Requirements regarding Confidentiality, Authentication and Integrity					
FCS_COP.1/DF-DES				X	Cryptographic Operation (DES)
FCS_COP.1/DF-AES				X	Cryptographic Operation (AES)
FCS_CKM.1/DF				X	Cryptographic key generation
FCS_CKM.4/DF				X	Cryptographic key destruction
FIA_UAU.2/DF				X	User authentication before any action
FIA_UAU.3/DF				X	Unforgeable authentication
FIA_UAU.5/DF				X	Multiple authentication mechanisms
FIA_UID.2/DF				X	User identification before any action
FPT_TDC.1/DF				X	Inter-TSF basic TSF data consistency
FTP_TRP.1/DF				X	Trusted path
Security Functional Requirements regarding Robustness					
FDP_ROL.1/DF				X	Basic rollback
FPR_UNL.1/DF				X	Unlinkability

SFR	TSF.Service	TSF.Protection	TSF.Control	TSF.DESFire	Description
FPT_RPL.1/DF				X	Replay detection
Security Functional Requirements regarding Secure Dynamic Messaging					
FDP_ETC.3/DF				X	Export of user data in unauthenticated state

7.2.2 TSF.Service

TSF.Service provides the following functionality:

TOE identification

FAU_SAS.1 is implemented by a test function that allows to store identification and/or pre-personalization data (including a unique ID for each die) for the TOE in the FLASH at the end of the tests in Phase 3.

Random Number Generation

The TOE provides a hardware (physical) random number generator (RNG) according to PTG.2 as described in [1]. The physical RNG comprises a hardware test functionality to detect faults in the circuitry of the RNG (total failure test). Therefore this functionality meets FCS_RNG.1/PTG2.

The TOE also provides a deterministic RNG according to DRG.3 as described in [1]. This functionality therefore meets FCS_RNG.1/DRG3.

7.2.3 TSF.Protection

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

Integrity protection of memories

As required by FDP_SDI.2, TSF.Protection supports the integrity of the ROM, RAM and Flash. The Flash is able to perform error correction. The ROM, RAM and Flash provide parity protection.

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

Protection against physical manipulations

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

The aspect of TSF.Protection is further supported by FPT_FLS.1 which controls the environmental conditions and triggers a reset in case these are out of bounds.

Logical protection

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

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

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

Cryptographic co-processors and cryptographic library

The cryptographic co-processors (TDES, AES) as well as the cryptographic library implements countermeasures against fault injection and information leakage. Another implemented mechanism to protect User Data from unwanted disclosure is an automatic clean-up of relevant registers after usage and before changing the TOE mode. Therefore, all FCS_COP.1 and FCS_CKM.4 iterations indirectly support TSF.Protection.

7.2.4 TSF.Control

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

Control of operating conditions

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

Mode control

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

TSF.Control provides access to the IC Dedicated Test Software in the Super System Mode before TOE delivery or to the IC Dedicated Support Software and Security IC Embedded Software after TOE delivery. It assures that it is not possible to enable access to the IC Dedicated Test Software after TOE delivery.

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

At the end of the wafer test the access to the IC Dedicated Test Software is disabled. TSF.Control ensures that it is not possible to switch back and reuse the test functions again. In addition, the test functions of the IC Dedicated Test Software require a

special sequence to execute a dedicated test routine. Therefore, TSF.Control limits the availability of the test functions as stated by FMT_LIM.2.

7.2.5 TSF.DESFire

TSF.DESFire provides the following functionality:

Authentication

This functionality 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 DES according to FIPS PUB 46-3 and for AES according to FIPS PUB 197. A hardware random number generator according to AIS31, functionality class PTG.2, and a deterministic random number generator seeded by the hardware random number generator (functionality DRG.3), are used to protect the authentication against attacks like e.g. replay. By this TSF.DESFire meets FCS_RNG.1/PTG2, FCS_RNG.1/DRG3, FCS_COP.1/DF-DES and FCS_COP.1/DF-AES.

This functionality also identifies the user to be authenticated by the currently selected context (card or specific application) and the key number. This meets FIA_UID.2/DF. The cryptographic authentication is used for the *Admin*, *AppMgr*, *DelAppMgr*, *AppUser*, *AppChangeUser*, *AppRollUser* and *OrigKeyUser*. The originality functionality which allows the authenticity of the TOE to be verified is performed by authenticating the *OrigKeyUser* using an *OriginalityKey*. Since the TOE can be used without authentication the "none" authentication is used to "authenticate" *Anybody*. Therefore it implements FIA_UAU.2/DF, FIA_UAU.5/DF and FMT_SMR.1/DF.

The authentication protocol requires the user to proof knowledge of a secret key by applying it on a freshly generated random challenge, generated to the TOE. This ensures that the authentication request itself cannot be forged or circumvented by attacks like replay or man-in-the-middle, therefore it meets FIA_UAU.3/DF and the relevant parts of FTP_TRP.1/DF and FPT_RPL.1/DF with respect to the authentication requests. Authentication of a user is initiated by an authentication request and the authentication state is reset if one of the following events occurs: Selecting an application or the card, Changing a key, Occurrence of any error during the execution of a command, starting a new authentication, deselection of the virtual card, Rolling a key set, Failed Proximity Check, Deleting an Application and Reset. By this FMT_SMF.1/DF is also implemented.

Access Control

This functionality 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) *AppUser* - to 4 different groups of operations on *Files*. The operations on *Files* 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. Therefore this functionality maintains the roles as required by FMT_SMR.1/DF.

Since this functionality also maintains the objects and Security Attributes as stated in the DESFire Access Control Policy, it also implements FDP_ACC.1/DF, FDP_ACF.1/DF and FMT_MSA.1/DF. Management of authentication data is necessary to separate the roles, therefore it also implements FMT_MTD.1/DF.

The primary use of the TOE is storage of data on behalf of the authorised users. The rules for data storage are defined by the DESFire Access Control Policy. The storage of data is an import of data with security attributes, therefore FDP_ITC.2/DF is also

implemented. This applies to the operations *File.Create* and *File.Delete* on the object *File* within *Applications*.

For the card the operations are *Application.Create* and *Application.Delete* on the object *Application*. If an *Application* is created default Security Attributes are assigned to the *Application*, thereby implementing FMT_MSA.3/DF. If an *Application* is deleted the keys used to authenticate the respective *AppMgr* and *AppUser* are destroyed. This implements FCS_CKM.4/DF.

The DESFire Access Control Policy also defines the rules for delegated-application support, which allows third party service providers to create their applications onto the issued TOE. After creation, a *DelApplication* has the same attributes as an *Application*.

This functionality also controls access to the security attributes. Because it also controls create and delete operations, it implements part of FMT_SMF.1/DF.

Finally the type consistency of the file types stored by the TOE is ensured. It ensures that values can not over- or underflow. Furthermore size limitations of files are obeyed. By this FPT_TDC.1/DF is implemented.

Encryption

TSF.DESFire provides a mechanism to protect the communication against eavesdropping by encryption. The encryption is requested by the file owner (i.e. the subject *AppUser* that has the right to perform *File.Change* on *File*) by setting an option in the attributes of *File*.

The encryption algorithm is the same as the one used during authentication for the session, however only the AES algorithm is supported, therefore it is bound to authentications with this algorithm. By this the functionality implements FCS_COP.1/DF-AES. The SFR FCS_CKM.1/DF generates the session keys used during the encryption. Note that the encryption functionality is active after an authentication is performed. If an authorised user sets the access control permissions in a way that an object is accessible to *Anybody* (refer to Access Control) this object can be accessed without authentication and therefore also without protection by this functionality.

TSF.DESFire 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, it 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 it can detect any injected/modified frame in the communication before the transfer of the encrypted frame.

The encryption for communication and the information to detect integrity violations implement FTP_TRP.1/DF with respect to the "confidentiality and/or data integrity verification for data transfers on request of the File owner".

When using the Secure Dynamic Messaging functionality, the TOE encrypts a configurable part of the File to be read when required by the File security attributes, therefore implementing FDP_ETC.3/DF.

Message Authentication Code

This functionality adds data to the communication stream that enables the terminal to detect integrity violations, replay attacks or man-in-the-middle attacks. Vice-versa it verifies the data sent by the terminal and returns an error code if such an attack is detected. It uses the cryptographic algorithm 128-bit AES CMAC. Only the AES algorithm is supported, therefore it is bound to authentications with this algorithm. This functionality

therefore implements FCS_COP.1/DF-AES. The SFR FCS_CKM.1/DF generates the session keys used during the calculation.

The detection mechanism covers all frames exchanged between the terminal and the card up to last frame with a MAC. Depending on the selected mode it can also detect what frame was injected/modified. By this FPT_RPL.1/DF is implemented.

The information to detect integrity violations implement FTP_TRP.1/DF with respect to the "confidentiality and/or data integrity verification for data transfers on request of the file owner".

When using the Secure Dynamic Messaging functionality, the TOE provides a mechanism for integrity protection for the File to be read when required by the File security attributes, therefore implementing FDP_ETC.3/DF.

Transaction

TSF.DESFire provides a transaction mechanism that 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. Note that it is possible to update files in up to 2 applications within one transaction.

This functionality 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.

The following reasons will abort a transaction: Selecting an application or the card, Changing a key, Occurrence of any error during the execution of a command, starting a new authentication, deselection of the virtual card, Rolling a key set, Failed Proximity Check, Deleting an Application and Reset. FDP_ROL.1/DF is therefore also implemented by this functionality.

Transaction Message Authentication Code

This functionality provides an option to the *Admin* or *AppMgr* and the *AppUser* to prove the authenticity of committed transactions on the TOE. In order to do this a MAC is calculated over a committed transaction. This MAC calculation is calculated by the use of the *AppTransactionMACKey* and is written to the *TransactionMAC* file. Both *AppTransactionMACKey* and *TransactionMAC* file can exist per application and need to be created by the *AppMgr* with *AppTransactionMACKey.Create*.

Note that only the AES encryption algorithm is supported. If an authorised user sets the access control permissions in a way that an object is accessible to *Anybody* (refer to Access Control) this object can be accessed without authentication and therefore also without protection by this functionality.

The information to detect integrity violations meets FTP_TRP.1/DF with respect to the "confidentiality and/or data integrity verification for data transfers on request of the file owner". Therefore this functionality also meets FTP_TRP.1/DF.

Preventing Traceabilty

This functionality provides an option to the Admin to use a random UID during ISO14443 anti-collision sequence. By this the card cannot be traced any more by simply retrieving its UID. Card specific information can be read out only by the *Admin*, *AppMgr*, *AppChangeUser*, *AppRollUser* and *AppUser* if this option is set.

The card specific information is protected and therefore FPR_UNL.1/DF is implemented. This functionality does not cover the data in the TOE file system. This data is protected by the DESFire Access Control Policy and the tracing protection depends on the access control configuration created by the authorised subjects.

Note that the TOE does also support the virtual card selection mechanism by which one of multiple virtual cards stored on one physical item can be selected. Although the TOE does support only one virtual card the selection mechanism is implemented in a way that unauthorised subjects cannot determine which virtual card (indicated by the virtual card ID) is supported by the TOE.

8 Bibliography

8.1 Evaluation documents

[1] A proposal for: Functionality classes for random number generators, Wolfgang Killmann, T-Systems GEI GmbH, Werner Schindler, Bundesamt für Sicherheit in der Informationstechnik (BSI), Version 2.0, 18 September 2011.

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

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

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

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

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

8.2 Developer documents

[7] MF3D(H)x3, MIFARE DESFire EV3 contactless smartcard IC, Product data sheet, DocStore number 487030, NXP Semiconductors, Revision 3.0, 4 May 2020.

[8] MF3D(H)x3C, MIFARE DESFire EV3C contactless smartcard IC, Product data sheet, DocStore number 588030, NXP Semiconductors, Revision 3.0, 4 May 2020.

[9] MF3D(H)x3, MIFARE DESFire EV3 Post Delivery Configuration, Preliminary data sheet addendum, DocStore number 581420, NXP Semiconductors, Revision 2.0, 6 March 2020.

[10] MF3D(H)x3, Wafer and Delivery Specification, Product data sheet addendum, DocStore number 580830, NXP Semiconductors, Revision 3.0, 15 May 2020.

[11] MF3D(H)x3, Information on Guidance and Operation, Guidance and Operation Manual, DocStore number 597610, NXP Semiconductors, Revision 1.0, 6 March 2020.

8.3 Standards

[12] FIPS PUB 197: Advanced Encryption Standard (AES), Federal Information Processing Standards Publication 197, US Department of Commerce/National Institute of Standards and Technology, 26 November 2001.

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

[14] NIST SP 800-38B: Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication, Morris Dworkin, National Institute of Standards and Technology, May 2005.

[15] NIST SP 800-67, Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher, revised January 2012, National Institute of Standards and Technology.

9 Legal information

9.1 Definitions

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

9.2 Disclaimers

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

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

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

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

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

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

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

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

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

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

Security — While NXP Semiconductors has implemented advanced security features, all products may be subject to unidentified vulnerabilities. Customers are responsible for the design and operation of their applications and products to reduce the effect of these vulnerabilities on customer's applications and products, and NXP Semiconductors accepts no liability for any vulnerability that is discovered. Customers should implement appropriate design and operating safeguards to minimize the risks associated with their applications and products.

9.3 Trademarks

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

MIFARE — is a trademark of NXP B.V.

DESFire — is a trademark of NXP B.V.

MIFARE Classic — is a trademark of NXP B.V.

Tables

Figures

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: 5 June 2020