STMicroelectronics

ST31 - K330A version F (dual or contactless mode only), with optional cryptographic library NESLIB 3.2, and optional technology MIFARE DESFire™ EV1 2.2
Security Target - Public Version

Common Criteria for IT security evaluation

SMD_MR31Zxxx_ST_13_002 Rev 01.02

September 2013

www.st.com

BLANK

1 Introduction

1.1 Security Target reference

1 Document identification: ST31 - K330A version F (dual or contactless mode only), with optional cryptographic library Neslib 3.2, and optional technology MIFARE DESFire™ EV1 2.2 SECURITY TARGET - PUBLIC VERSION.

2 Version number: Rev 01.02, issued September 2013.

3 Registration: registered at ST Microelectronics under number SMD_MR31Zxxx_ST_13_002.

1.2 Purpose

4 This document presents **the Security Target - Public version (ST)** of the **ST31 - K330A** Security Integrated Circuits (IC), designed on the **ST31 platform of STMicroelectronics**, with Dedicated Software (DSW), optional cryptographic library **Neslib 3.2,** and optional technology **MIFARE DESFire™ EV1 2.2**.

5 The precise reference of the Target of Evaluation (TOE) and the security IC features are given in *Section 3: TOE description*.

6 A glossary of terms and abbreviations used in this document is given in *Appendix A: Glossary*.

Contents

List of tables

List of figures

2 Context

7 The Target of Evaluation (TOE) referred to in *Section 3: TOE description*, is evaluated under the French IT Security Evaluation and Certification Scheme and is developed by the Secure Microcontrollers Division of STMicroelectronics (ST).

8 The assurance level of the performed Common Criteria (CC) IT Security Evaluation is EAL 5 augmented by ALC_DVS.2 and AVA_VAN.5.

9 The intent of this Security Target is to specify the Security Functional Requirements (SFRs) and Security Assurance Requirements (SARs) applicable to the TOE security ICs, and to summarise their chosen TSF services and assurance measures.

10 This ST claims to be an instantiation of the "*Security IC Platform Protection Profile*" (PP) registered and certified under the reference *BSI-PP-0035* in the German IT Security Evaluation and Certification Scheme, ***with the following augmentations***:

- Addition #1: “Support of Cipher Schemes” from *AUG*
- Addition #4: “Area based Memory Access Control” from *AUG*
- Additions specific to this Security Target.

The original text of this PP is typeset as indicated here, its augmentations from *AUG* as indicated here, when they are reproduced in this document.

11 Extensions introduced in this ST to the SFRs of the Protection Profile (PP) are **exclusively** drawn from the Common Criteria part 2 standard SFRs.

12 This ST makes various refinements to the above mentioned PP and *AUG*. They are all properly identified in the text typeset as ***indicated here***. The original text of the PP is repeated as scarcely as possible in this document for reading convenience. All PP identifiers have been however prefixed by their respective origin label: ***BSI*** for *BSI-PP-0035*, ***AUG1*** for Addition #1 of *AUG* and ***AUG4*** for Addition #4 of *AUG*.

3 TOE description

3.1 TOE identification

13 The Target of Evaluation (TOE) is the ST31 - K330A version F (dual or contactless mode only), with the optional cryptographic library Neslib 3.2, and/or the optional library MIFARE DESFire™ EV1 2.2, with guidance documentation.

Table 1. TOE identification

IC Maskset name	Maskset Major version	IC version	Master identification number (1)	OST name(1)	OST version (1)	Optional crypto library name & version(2)	Optional MIFARE DESFire EV1 version(3)
K330A	A	F	0033h	YGD	0013h	Neslib 3.2 1320h	2.2

1. Part of the product information.
2. See the Neslib User Manual referenced in *Section 9* .
3. See the MIFARE DESFire EV1 User Manual referenced in *Section 9* .

14 The IC maskset name is the product hardware identification.
The maskset major version is updated when the full maskset is changed (i.e. all layers of the maskset are changed at the same time).
The IC version is updated for any change in hardware (i.e. part of the layers of the maskset) or in the OST software.
The maskset name with major version and IC version (i.e. K330A version F) fully identify the IC (hardware and OST).

15 The K330A version F, is dedicated to Dual mode or contactless mode only. The 1.8V voltage range is deactivated.

16 Different derivative devices may be configured by ST during the manufacturing or packaging process, depending on the customer needs. They all share the same hardware design and the same maskset.

17 The configuration of the derivative devices can impact the I/O mode, the available NVM memory size, the availability of Nescrypt and the availability of MIFARE support features, as detailed here below:

Table 2. Derivative devices configuration possibilities

Features	Possible values
I/O mode	Dual mode, Contactless only
NVM size	52 Kbytes, 38 Kbytes, 22 Kbytes, 16 Kbytes
Nescrypt	Active, Inactive
MIFARE support	Active, Inactive

18 All combinations of different features values are possible and covered by this certification. All possible configurations can vary under a unique IC (i.e. K330A), and without impact on security.

19 The Master identification number is unique for all product configurations.
Each derivative device has a specific Child product identification number, also part of the product information, and specified in the Data Sheet, referenced in *Section 9*.

20 All along the product life, the marking on the die, a set of accessible registers and a set of specific instructions allow the customer to check the product information, providing the identification elements, as listed in *Table 1: TOE identification*, and the configuration elements as detailed in the Data Sheet, referenced in *Section 9*.

21 In this Security Target, the term "DESFire" means MIFARE DESFire™ EV1 2.2.

22 The rest of this document applies to all possible configurations of the TOE, with or without Neslib, or DESFire libraries, except when a restriction is mentioned. For easier reading, the restrictions are typeset as indicated here.

3.2 TOE overview

23 Designed for secure ID and banking applications, the TOE is a serial access microcontroller that incorporates the most recent generation of ARM® processors for embedded secure systems. Its SecurCore® SC000™ 32-bit RISC core is built on the Cortex™ M0 core with additional security features to help to protect against advanced forms of attacks.

24 The TOE features hardware accelerators for advanced cryptographic functions, with built-in countermeasures against side channel attacks. The AES (Advanced Encryption Standard) accelerator provides a high-performance implementation of AES-128, AES-192 and AES-256 algorithms. The 3-key triple DES accelerator (EDES+) supports efficiently the Data Encryption Standard (DES *[2]*), enabling Cipher Block Chaining (CBC) mode, fast DES and triple DES computation. If Nescrypt is active, the NESCRYPT crypto-processor allows fast and secure implementation of the most popular public key cryptosystems with a high level of performance (*[4]*, *[8]*, *[12]*, *[18]*,*[19]*, *[20]*, *[21]*).

As randomness is a key stone in many applications, the ST31 - K330A features a highly reliable True Random Number Generator (TRNG), compliant with P2 Class of AIS31 *[1]* and directly accessible thru dedicated registers.

This device also includes the ARM® SecurCore® SC000™ memory protection unit (MPU), which enables the user to define its own region organization with specific protection and access permissions.

25 The TOE offers a contact serial communication interface fully compatible with the ISO/IEC 7816-3 standard, and a contactless interface including an RF Universal Asynchronous Receiver Transmitter (RF UART), enabling communication up to 848 Kbits/s compatible with the ISO/IEC 14443 Type A, B and B', and PayPass™ standards.
These interfaces can be used simultaneously (dual mode), or the contact interface can be deactivated (see *Table 2: Derivative devices configuration possibilities*).

26 In a few words, the ST31 - K330A offers a unique combination of high performances and very powerful features for high level security:

- Die integrity,
- Monitoring of environmental parameters,
- Protection mechanisms against faults,
- Hardware Security Enhanced DES accelerator,
- True Random Number Generator,
- ISO 3309 CRC calculation block,
- Memory Protection Unit,
- optional NExt Step CRYPTography accelerator (NESCRYPT),
- optional cryptographic library,
- optional secure MIFARE DESFire EV1 library.

27 The TOE includes in the ST protected ROM a Dedicated Software which provides full test capabilities (operating system for test, called "OST"), not accessible by the Security IC Embedded Software (ES), after TOE delivery.

28 The Security IC Embedded Software (ES) is in User ROM.

The ES is not part of the TOE and is out of scope of the evaluation, except Neslib and DESFire, when they are embedded.

29 The TOE optionally comprises a specific application in User ROM: this applicative Embedded Software is a cryptographic library called Neslib. Neslib is a cutting edge cryptographic library in terms of security and performance.

Neslib is embedded by the ES developper in his applicative code.
Note that Neslib can only be used if Nescrypt is active.

Neslib provides the most useful operations in public key algorithms and protocols:

- an asymmetric key cryptographic support module, supporting secure modular arithmetic with large integers, with specialized functions for Rivest, Shamir & Adleman Standard cryptographic algorithm (RSA *[20]*),
- an asymmetric key cryptographic support module that provides very efficient basic functions to build up protocols using Elliptic Curves Cryptography on prime fields GF(p) *[18]*,
- an asymmetric key cryptographic support module that provides secure hash functions (SHA-1, SHA-224, SHA-256, SHA-384, and SHA-512 *[4]*),
- a symmetric key cryptographic support module whose base algorithm is the Advanced Encryption Standard cryptographic algorithm (AES *[7]*),
- prime number generation *[6]*.

30 The TOE optionally comprises a specific application in User ROM: this applicative Embedded Software is a MIFARE technology library.
This library is a secure library called MIFARE DESFire™ EV1. DESFire features a mutual three pass authentication, a data encryption on RF channel, and a flexible self-securing file system.
DESFire is embedded by the ES developper in his applicative code.
Note that DESFire can only be used if Nescrypt and MIFARE support are active.

31 The user guidance documentation, part of the TOE, consists of:

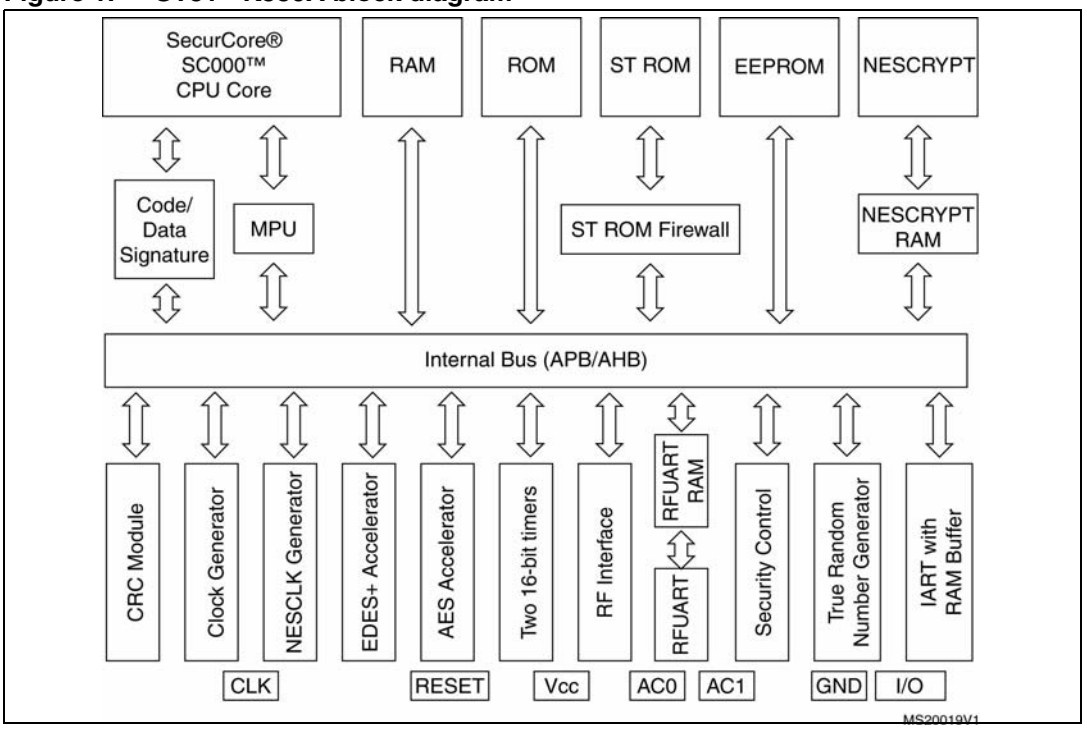
- the product Data Sheet and die description,
- the product family Security Guidance,
- the AIS31 user manuals,
- the product family programming manual,
- the ARM SC000 Technical Reference Manual,
- optionally the Neslib user manual,
- optionally the MIFARE DESFire EV1 user manual.

32 The complete list of guidance documents is detailed in *Section 9*.

33 In addition, the ROM of the tested samples contains an operating system called "Card Manager" that allows the evaluators to use a set of commands with the I/O or in RF mode, and to load in EEPROM (or in RAM) test software. The card manager is not part of the TOE, and not in the scope of this evaluation. It will not be present on the field (Phase 7).

34 *Figure 1* provides an overview of the ST31 - K330A.

Figure 1. ST31 - K330A block diagram



3.3 TOE life cycle

35 This Security Target is fully conform to the claimed PP. In the following, just a summary and some useful explanations are given. For complete details on the TOE life cycle, please refer to the *Security IC Platform Protection Profile* (*BSI-PP-0035*), section 1.2.3.

36 The composite product life cycle is decomposed into 7 phases. Each of these phases has the very same boundaries as those defined in the claimed protection profile.

37 The life cycle phases are summarized in *Table 3*.

38 The limit of the evaluation corresponds to phases 2, 3 and optionally 4, including the delivery and verification procedures of phase 1, and the TOE delivery either to the IC packaging manufacturer or to the composite product integrator ; procedures corresponding to phases 1, 5, 6 and 7 are outside the scope of this evaluation.

39 In the following, the term "Composite product manufacturing" is uniquely used to indicate phases 1, optionally 4, 5 and 6 all together.
This ST also uses the term "Composite product manufacturer" which includes all roles responsible of the TOE during phases 1, optionally 4, 5 and 6.

40 The TOE is delivered after Phase 3 in form of wafers or after Phase 4 in packaged form, depending on the customer's order.

41 In the following, the term "TOE delivery" is uniquely used to indicate:

- after Phase 3 (or before Phase 4) if the TOE is delivered in form of wafers or sawn wafers (dice) or
- after Phase 4 (or before Phase 5) if the TOE is delivered in form of packaged products.

42 The TOE is delivered in USER configuration.

Table 3. Composite product life cycle phases

Phase	Name	Description	Responsible party
1	IC embedded software development	security IC embedded software development specification of IC pre-personalization requirements	IC embedded software developer
2	IC development	IC design IC dedicated software development	IC developer: **ST**
3	IC manufacturing	integration and photomask fabrication IC production IC testing pre-personalisation	IC manufacturer: **ST**
4	IC packaging	security IC packaging (and testing) pre-personalisation if necessary	IC packaging manufacturer: **ST** or **NEDCARD** or **SMARTFLEX**
5	Composite product integration	composite product finishing process composite product testing	Composite product integrator
6	Personalisation	composite product personalisation composite product testing	Personaliser
7	Operational usage	composite product usage by its issuers and consumers	End-consumer

3.4 TOE environment

43 Considering the TOE, three types of environments are defined:

- Development environment corresponding to phase 2,
- Production environment corresponding to phase 3 and optionally 4,
- Operational environment, including phase 1 and from phase 4 or 5 to phase 7.

3.4.1 TOE Development Environment

44 To ensure security, the environment in which the development takes place is secured with controllable accesses having traceability. Furthermore, all authorised personnel involved fully understand the importance and the strict implementation of defined security procedures.

45 The development begins with the TOE's specification. All parties in contact with sensitive information are required to abide by Non-Disclosure Agreements.

46 Design and development of the IC then follows, together with the dedicated and engineering software and tools development. The engineers use secure computer systems (preventing unauthorised access) to make their developments, simulations, verifications and generation of the TOE's databases. Sensitive documents, files and tools, databases on tapes, and printed circuit layout information are stored in appropriate locked cupboards/safe. Of paramount importance also is the disposal of unwanted data (complete electronic erasures) and documents (e.g. shredding).

47 The development centres involved in the development of the TOE are the following: **ST ROUSSET (FRANCE)** and **ST ANG MO KIO (SINGAPORE)**, for the design activities, **ST ROUSSET (FRANCE)**, for the engineering activities, **ST ROUSSET (FRANCE)** and **ST ZAVENTEM (BELGIUM)** for the software development activities.

48 Reticules and photomasks are generated from the verified IC databases; the former are used in the silicon Wafer-fab processing. As reticules and photomasks are generated off-site, they are transported and worked on in a secure environment with accountability and traceability of all (good and bad) products. During the transfer of sensitive data electronically, procedures are established to ensure that the data arrive only at the destination and are not accessible at intermediate stages (e.g. stored on a buffer server where system administrators make backup copies).

49 The authorized sub-contractors involved in the TOE mask manufacturing can be **DNP (JAPAN)** and **DPE (ITALY)**.

3.4.2 TOE production environment

50 As high volumes of product commonly go through such environments, adequate control procedures are necessary to account for all product at all stages of production.

51 Production starts within the Wafer-fab; here the silicon wafers undergo the diffusion processing. Computer tracking at wafer level throughout the process is commonplace. The wafers are then taken into the test area. Testing of each TOE occurs to assure conformance with the device specification.

52 The authorized front-end plant involved in the manufacturing of the TOE is **ST ROUSSET (FRANCE)**.

53 The authorized EWS plant involved in the testing of the TOE can be **ST ROUSSET (FRANCE)** or **ST TOA PAYOH (SINGAPORE)**.

54 Wafers are then scribed and broken such as to separate the functional from the non-functional ICs. The latter is discarded in a controlled accountable manner. The good ICs are then packaged in phase 4, in a back-end plant. When testing, programming or deliveries are done offsite, ICs are transported and worked on in a secure environment with accountability and traceability of all (good and bad) products.

55 When the product is delivered after phase 4, the authorized back-end plant involved in the packaging of the TOE can be **ST BOUSKOURA (MOROCCO)** or **ST CALAMBA (THE PHILIPPINES)** or **NEDCARD (THE NETHERLANDS)** or **SMARTFLEX (SINGAPORE)**.

56 The other sites that can be involved during the production of the TOE are **ST LOYANG (SINGAPORE)** for the logistics, and **ST SHENZEN (CHINA)** or **DISCO (GERMANY)** for the wafers backlap and sawing.

3.4.3 TOE operational environment

57 A TOE operational environment is the environment of phases 1, optionally 4, then 5 to 7.

58 At phases 1, 4, 5 and 6, the TOE operational environment is a controlled environment.

59 End-user environments (phase 7): composite products are used in a wide range of applications to assure authorised conditional access. Examples of such are pay-TV, banking cards, brand protection, portable communication SIM cards, health cards, transportation cards, access management, identity and passport cards. The end-user environment therefore covers a wide range of very different functions, thus making it difficult to avoid and monitor any abuse of the TOE.

4 Conformance claims

4.1 Common Criteria conformance claims

60 The ST31 - K330A Security Target claims to be conformant to the Common Criteria version 3.1 revision 4.

61 Furthermore it claims to be CC Part 2 (*CCMB-2012-09-002*) extended and CC Part 3 (*CCMB-2012-09-003*) conformant. The extended Security Functional Requirements are those defined in the *Security IC Platform Protection Profile* (*BSI-PP-0035*).

62 The assurance level for the ST31 - K330A Security Target is ***EAL 5*** augmented by ALC_DVS.2 and AVA_VAN.5.

4.2 PP Claims

4.2.1 PP Reference

63 The ST31 - K330A Security Target claims strict conformance to the *Security IC Platform Protection Profile* (*BSI-PP-0035*), as required by this Protection Profile.

4.2.2 PP Refinements

64 The main refinements operated on the *BSI-PP-0035* are:

- Addition #1: “Support of Cipher Schemes” from *AUG*,
- Addition #4: “Area based Memory Access Control” from *AUG*,
- Specific additions for DESFire,
- Refinement of assurance requirements.

65 All refinements are indicated with type setting text ***as indicated here***, original text from the *BSI-PP-0035* being typeset as indicated here. Text originating in *AUG* is typeset as indicated here.

4.2.3 PP Additions

66 The security environment additions relative to the PP are summarized in *Table 4*.

67 The additional security objectives relative to the PP are summarized in *Table 5*.

68 A simplified presentation of the TOE Security Policy (TSP) is added.

69 The additional SFRs for the TOE relative to the PP are summarized in *Table 7*.

70 The additional SARs relative to the PP are summarized in *Table 10*.

4.2.4 PP Claims rationale

71 The differences between this Security Target security objectives and requirements and those of *BSI-PP-0035*, to which conformance is claimed, have been identified and justified in *Section 6* and in *Section 7*. They have been recalled in the previous section.

72 In the following, the statements of the security problem definition, the security objectives, and the security requirements are consistent with those of the *BSI-PP-0035*.

73 The security problem definition presented in *Section 5*, clearly shows the additions to the security problem statement of the PP.

74 The security objectives rationale presented in *Section 6.3* clearly identifies modifications and additions made to the rationale presented in the *BSI-PP-0035*.

75 Similarly, the security requirements rationale presented in *Section 7.4* has been updated with respect to the protection profile.

76 All PP requirements have been shown to be satisfied in the extended set of requirements whose completeness, consistency and soundness have been argued in the rationale sections of the present document.

5 Security problem definition

77 This section describes the security aspects of the environment in which the TOE is intended to be used and addresses the description of the assets to be protected, the threats, the organisational security policies and the assumptions.

78 Note that the origin of each security aspect is clearly identified in the prefix of its label. Most of these security aspects can therefore be easily found in the *Security IC Platform Protection Profile* (*BSI-PP-0035*), section 3. Only those originating in *AUG*, and the ones introduced in this Security Target, are detailed in the following sections.

79 A summary of all these security aspects and their respective conditions is provided in *Table 4*.

5.1 Description of assets

80 Since this Security Target claims strict conformance to the *Security IC Platform Protection Profile* (*BSI-PP-0035*), the assets defined in section 3.1 of the Protection Profile are applied and the assets regarding threats are clarified in this Security Target.

81 The assets regarding the threats are:

- 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 and product in any form,
- the TOE correct operation,
- the Security IC Embedded Software, stored and in operation,
- the security services provided by the TOE for the Security IC Embedded Software,
- the cryptographic co-processors for Triple-DES and AES, the random number generator,
- when DESFire is embedded, the special functions for the communication with an external interface device,
- the User Data comprising, especially when DESFire is embedded,
  - authentication data like keys,
  - issuer data like card holder name or processing options,
  - representation of monetary values, e.g. a stored value for transport applications,
- the TSF Data.

82 This Security Target includes optionally Security IC Embedded Software and therefore does contain more assets compared to *BSI-PP-0035*. These assets are described above.

Table 4. Summary of security environment

	Label	Title
TOE threats	BSI.T.Leak-Inherent	Inherent Information Leakage
	BSI.T.Phys-Probing	Physical Probing
	BSI.T.Malfunction	Malfunction due to Environmental Stress
	BSI.T.Phys-Manipulation	Physical Manipulation
	BSI.T.Leak-Forced	Forced Information Leakage
	BSI.T.Abuse-Func	Abuse of Functionality
	BSI.T.RND	Deficiency of Random Numbers
	AUG4.T.Mem-Access	Memory Access Violation
	T.Data_Modification	Unauthorised data modification
	T.Impersonate	Impersonating authorised users during authentication
	T.Cloning	Cloning
	T.Confid-Applic-Code	DESFire code confidentiality
	T.Confid-Applic-Data	DESFire data confidentiality
	T.Integ-Applic-Code	DESFire code integrity
	T.Integ-Applic-Data	DESFire data integrity
	T.Resource	DESFire resource unavailability
OSPs	BSI.P.Process-TOE	Protection during TOE Development and Production
	AUG1.P.Add-Functions	Additional Specific Security Functionality (Cipher Scheme Support)
	P.Confidentiality	Confidentiality during communication
	P.Transaction	Transaction mechanism
	P.No-Trace	Un-traceability of end-users
	P.Plat-Appl	Usage of hardware platform
	P.Resp-Appl	Treatment of user data
Assumptions	BSI.A.Process-Sec-IC	Protection during Packaging, Finishing and Personalisation
	BSI.A.Plat-Appl	Usage of Hardware Platform
	BSI.A.Resp-Appl	Treatment of User Data
	A.Secure-Values	Usage of secure values
	A.Terminal-Support	Terminal support to ensure integrity and confidentiality

5.2 Threats

83 The threats are described in the *BSI-PP-0035*, section 3.2. Only those originating in *AUG* and those related to DESFire are detailed in the following section.

BSI.T.Leak-Inherent Inherent Information Leakage

BSI.T.Phys-Probing	Physical Probing
BSI.T.Malfunction	Malfunction due to Environmental Stress
BSI.T.Phys-Manipulation	Physical Manipulation
BSI.T.Leak-Forced	Forced Information Leakage
BSI.T.Abuse-Func	Abuse of Functionality
BSI.T.RND	Deficiency of Random Numbers
AUG4.T.Mem-Access	Memory Access Violation: Parts of the ***Security IC*** Embedded Software may cause security violations by accidentally or deliberately accessing restricted data (which may include code). Any restrictions are defined by the security policy of the specific application context and must be implemented by the ***Security IC*** Embedded Software. Clarification: This threat does not address the proper definition and management of the security rules implemented by the Security IC Embedded Software, this being a software design and correctness issue. This threat addresses the reliability of the abstract machine targeted by the software implementation. To avert the threat, the set of access rules provided by this TOE should be undefeated if operated according to the provided guidance. The threat is not realized if the Security IC Embedded Software is designed or implemented to grant access to restricted information. It is realized if an implemented access denial is granted under unexpected conditions or if the execution machinery does not effectively control a controlled access. Here the attacker is expected to (i) take advantage of flaws in the design and/or the implementation of the TOE memory access rules (refer to BSI.T.Abuse-Func but for functions available after TOE delivery), (ii) introduce flaws by forcing operational conditions (refer to BSI.T.Malfunction) and/or by physical manipulation (refer to BSI.T.Phys-Manipulation). This attacker is expected to have a high level potential of attack.

84 The following additional threats are related to DESFire. They are valid in case DESFire is embedded in the TOE.

	Unauthorised data modification:
T.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.
	Impersonating authorised users during authentication:
T.Impersonate	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.
	Cloning:
T.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.

T.Confid-Applic-Code	DESFire code confidentiality: MIFARE DESFire EV1 Licensed product code must be protected against unauthorized disclosure. This relates to attacks at runtime to gain read or compare access to memory area where the MIFARE DESFire EV1 licensed product executable code is stored. The attacker executes an application to disclose code belonging to MIFARE DESFire EV1 Licensed product.
T.Confid-Applic-Data	DESFire data confidentiality: MIFARE DESFire EV1 Licensed product data must be protected against unauthorized disclosure. This relates to attacks at runtime to gain read or compare access to the MIFARE DESFire EV1 licensed product data by another application. For example, the attacker executes an application that tries to read data belonging to MIFARE DESFire EV1 Licensed product.
T.Integ-Applic-Code	DESFire code integrity: MIFARE DESFire EV1 Licensed product code must be protected against unauthorized modification. This relates to attacks at runtime to gain write access to memory area where the MIFARE DESFire EV1 licensed product executable code is stored. The attacker executes an application that tries to alter (part of) the DESFire EV1 code.
T.Integ-Applic-Data	DESFire data integrity: MIFARE DESFire EV1 Licensed product data must be protected against unauthorized modification. This relates to attacks at runtime to gain write access to the MIFARE DESFire EV1 Licensed product data by another application. The attacker executes an application that tries to alter (part of) the DESFire EV1 Licensed product data.
T.Resource	DESFire resource unavailability: The availability of resources for the MIFARE DESFire EV1 Licensed product shall be controlled to prevent denial of service or malfunction. An attacker prevents correct execution of DESFire EV1 through consumption of some resources of the card: e.g. RAM or non volatile RAM.

5.3 Organisational security policies

85 The TOE provides specific security functionality that can be used by the ***Security IC*** Embedded 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 ***Security IC*** application, against which threats the ***Security IC*** Embedded Software will use the specific security functionality.

86 ST applies the Protection policy during TOE Development and Production (*BSI.P.Process-TOE*) as specified below.

87 ***ST*** applies the Additional Specific Security Functionality policy (*AUG1.P.Add-Functions*) as specified below.

88 New Organisational Security Policies (OSPs) are defined here below:

89 P.Confidentiality, P.Transaction and P.No-Trace are related to DESFire, and valid in case DESFire is embbeded in the TOE.

90 P.Plat-Appl and P.Resp-Appl are related to the ES that is part of the evaluation (Neslib and/or DESFire), and valid in case Neslib or DESFire are embbeded in the TOE.

BSI.P.Process-TOE	Protection during TOE Development and Production: An accurate identification ***is*** established for the TOE. This requires that each instantiation of the TOE carries this unique identification.
AUG1.P.Add-Functions	Additional Specific Security Functionality: The TOE shall provide the following specific security functionality to the Security IC Embedded Software: – Data Encryption Standard (DES), – Triple Data Encryption Standard (3DES), – Advanced Encryption Standard (AES), – ***Elliptic Curves Cryptography on GF(p),*** if Neslib is embedded only***,*** – ***Secure Hashing (SHA-1, SHA-224, SHA-256, SHA-384, SHA-512),*** if Neslib is embedded only***,*** – Rivest-Shamir-Adleman (RSA)***,*** if Neslib is embedded only, – ***Prime Number Generation,*** if Neslib is embedded only***.*** Note that DES is no longer recommended as an encryption function in the context of smart card applications. Hence, Security IC Embedded Software may need to use triple DES to achieve a suitable strength.
P.Confidentiality	Confidentiality during communication: The TOE shall provide the possibility to protect selected data elements from eavesdropping during contact-less communication. The TOE shall also provide the possibility to detect replay or man-in-the-middle attacks within a session.
P.Transaction	Transaction mechanism: The TOE shall provide the possibility to combine a number of data modification operations in one transaction, so that either all operations or no operation at all is performed.
P.No-Trace	Un-traceability of end-users: The TOE shall provide the ability that authorised subjects can prevent that end-user of TOE may be traced by unauthorised subjects without consent. Tracing of end-users may happen by performing a contact-less communication with the TOE when the end-user is not aware of it. Typically this involves retrieving the UID or any freely accessible data element.
P.Plat-Appl	Usage of hardware platform: The Security IC Embedded Software, part of the TOE, uses the TOE hardware platform according to the assumption A.Plat-Appl defined in *BSI-PP-0035*.
P.Resp-Appl	Treatment of user data: The Security IC Embedded Software, part of the TOE, treats user data according to the assumption A.Resp-Appl defined in *BSI-PP-0035*.

5.4 Assumptions

91 The following assumptions are described in the *BSI-PP-0035*, section 3.4.

BSI.A.Process-Sec-IC	Protection during Packaging, Finishing and Personalisation

BSI.A.Plat-Appl	Usage of Hardware Platform
BSI.A.Resp-Appl	Treatment of User Data

92 The following assumptions are defined for DESFire only.
Thus, they do not contradict with the security problem definition of the *BSI-PP-0035*, as they are only related to assets which are out of the scope of this PP.

93 In consequence, the addition of these asumptions does not contradict with the strict conformance claim on the *BSI-PP-0035*.

94 These assumptions are valid in case DESFire is embedded in the TOE.

A.Secure-Values	Usage of secure values: Only confidential and secure keys shall be used to set up the authentication and access rights in DESFire. These values are generated outside the TOE and they are downloaded to the TOE.
A.Terminal-Support	Terminal support to ensure integrity and confidentiality: The terminal verifies information sent by the TOE in order to ensure integrity and confidentiality of the communication.

6 Security objectives

95 The security objectives of the TOE cover principally the following aspects:

- integrity and confidentiality of assets,
- protection of the TOE and associated documentation during development and production phases,
- provide random numbers,
- provide cryptographic support and access control functionality.

96 A summary of all security objectives is provided in *Table 5*.

97 Note that the origin of each objective is clearly identified in the prefix of its label. Most of these security aspects can therefore be easily found in the protection profile. Only those originating in *AUG*, and the one introduced in this Security Target, are detailed in the following sections.

Table 5. Summary of security objectives

	Label	Title
TOE	BSI.O.Leak-Inherent	Protection against Inherent Information Leakage
	BSI.O.Phys-Probing	Protection against Physical Probing
	BSI.O.Malfunction	Protection against Malfunctions
	BSI.O.Phys-Manipulation	Protection against Physical Manipulation
	BSI.O.Leak-Forced	Protection against Forced Information Leakage
	BSI.O.Abuse-Func	Protection against Abuse of Functionality
	BSI.O.Identification	TOE Identification
	BSI.O.RND	Random Numbers
	AUG1.O.Add-Functions	Additional Specific Security Functionality
	AUG4.O.Mem-Access	***Dynamic*** Area based Memory Access Control
	O.Access-Control	Access Control for DESFire
	O.Authentication	Authentication for DESFire
	O.Confidentiality	DESFire Confidential Communication
	O.Type-Consistency	DESFire Data type consistency
	O.Transaction	DESFire Transaction mechanism
	O.No-Trace	Preventing Traceability for DESFire
	O.Plat-Appl	Usage of hardware platform
	O.Resp-Appl	Treatment of user data
	O.Resource	Resource availability for DESFire
	O.Firewall	DESFire firewall
	O.Shr-Res	DESFire data cleaning for resource sharing
	O.Verification	DESFire code integrity check

Table 5. Summary of security objectives (continued)

	Label	Title
Environments	BSI.OE.Plat-Appl	Usage of Hardware Platform
	BSI.OE.Resp-Appl	Treatment of User Data
	BSI.OE.Process-Sec-IC	Protection during composite product manufacturing
	OE.Secure-Values	Generation of secure values
	OE.Terminal-Support	Terminal support to ensure integrity and confidentiality

6.1 Security objectives for the TOE

BSI.O.Leak-Inherent — Protection against Inherent Information Leakage

BSI.O.Phys-Probing — Protection against Physical Probing

BSI.O.Malfunction — Protection against Malfunctions

BSI.O.Phys-Manipulation — Protection against Physical Manipulation

BSI.O.Leak-Forced — Protection against Forced Information Leakage

BSI.O.Abuse-Func — Protection against Abuse of Functionality

BSI.O.Identification — TOE Identification

BSI.O.RND — Random Numbers

AUG1.O.Add-Functions — Additional Specific Security Functionality:
The TOE must provide the following specific security functionality to the ***Security IC*** Embedded Software:
– Data Encryption Standard (DES),
– Triple Data Encryption Standard (3DES),
– Advanced Encryption Standard (AES), if Neslib is embedded only,
– ***Elliptic Curves Cryptography on GF(p),*** if Neslib is embedded only,
– ***Secure Hashing (SHA-1, SHA-224, SHA-256, SHA-384, SHA-512),*** if Neslib is embedded only,
– Rivest-Shamir-Adleman (RSA), if Neslib is embedded only,
– ***Prime Number Generation,*** if Neslib is embedded only.

AUG4.O.Mem-Access — ***Dynamic*** Area based Memory Access Control:
The TOE must provide the ***Security IC*** Embedded Software with the capability to define ***dynamic memory segmentation and protection***. The TOE must then enforce ***the defined access rules*** so that access of software to memory areas is controlled as required, for example, in a multi-application environment.

98 The following objectives are only valid in case DESFire is embedded:

O.Access-Control	Access Control for DESFire: 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 for DESFire: 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.Confidentiality	DESFire 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 element. The TOE shall also provide the possibility to detect replay or man-in-the-middle attacks within a session. This shall be implemented by checking verification data sent by the terminal and providing verification data to the terminal.
O.Type-Consistency	DESFire Data type consistency: The TOE must provide a consistent handling of the different supported data types. This comprises over- and underflow checking for values, for data file sizes and record handling.
O.Transaction	DESFire Transaction mechanism: The TOE must be able to provide a transaction mechanism that allows to update multiple data elements either all in common or none of them.
O.No-Trace	Preventing Traceability for DESFire: The TOE must be able to prevent that the TOE end-user can be traced. This shall be done by providing an option that disables the transfer of any information that is suitable for tracing an end-user by an unauthorised subject.
O.Plat-Appl	Usage of hardware platform: To ensure that the TOE is used in a secure manner the Security IC Embedded Software, part of the TOE, shall be designed so that the requirements from the following documents are met: (i) hardware data sheet for the TOE, (ii) data sheet of the IC dedicated software of the TOE, (iii) TOE application notes, other guidance documents, and (iii) findings of the TOE evaluation reports relevant for the Security IC Embedded Software.
O.Resp-Appl	Treatment of user data: Security relevant User Data (especially cryptographic keys) are treated by the Security IC Embedded Software as required by the security needs of the specific application context. For example the Security IC Embedded Software will not disclose security relevant user data to unauthorised users or processes when communicating with a terminal.

O.Resource	Resource availability for DESFire: The TOE shall control the availability of resources for MIFARE DESFire EV1 Licensed product.
O.Firewall	DESFire firewall: The TOE shall ensure isolation of data and code between MIFARE DESFire EV1 and the other applications. An application shall not read, write, compare any piece of data or code belonging to the MIFARE DESFire EV1 Licensed product.
O.Shr-Res	DESFire data cleaning for resource sharing: It shall be ensured that any hardware resource, that is shared by MIFARE DESFire EV1 and other applications or by any application which has access to such hardware resource, is always cleaned (using code that is part of the MIFARE DESFire EV1 system and its certification) whenever MIFARE DESFire EV1 is interrupted by the operation of another application. The only exception is buffers as long as these buffers do not contain other information than what is communicated over the contactless interface or has a form that is no different than what is normally communicated over the contacless interface. For example, no data shall remain in a hardware crytographic coprocessor when MIFARE DESFire EV1 is interrupted by another application.
O.Verification	DESFire code integrity check: The TOE shall ensure that MIFARE DESFire EV1 code is verified for integrity and authenticity prior being executed.

6.2 Security objectives for the environment

99 Security Objectives for the Security IC Embedded Software development environment (phase 1):

BSI.OE.Plat-Appl	Usage of Hardware Platform
BSI.OE.Resp-Appl	Treatment of User Data

100 Security Objectives for the operational Environment (phase 4 up to 6):

BSI.OE.Process-Sec-IC	Protection during composite product manufacturing

101 This section details the security objectives for the operational environment, related to DESFire, and to be enforced after TOE delivery up to phase 6.

102 The following security objectives for the operational environment are only valid if DESFire is embedded in the TOE:

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

OE.Terminal-Support	Terminal support to ensure integrity and confidentiality: 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.

6.3 Security objectives rationale

103 The main line of this rationale is that the inclusion of all the security objectives of the *BSI-PP-0035* protection profile, together with those in *AUG*, and those introduced in this ST, guarantees that all the security environment aspects identified in *Section 5* are addressed by the security objectives stated in this chapter.

104 Thus, it is necessary to show that:

- security environment aspects from *AUG*, and from this ST, are addressed by security objectives stated in this chapter,
- security objectives from *AUG*, and from this ST, are suitable (i.e. they address security environment aspects),
- security objectives from *AUG*, and from this ST, are consistent with the other security objectives stated in this chapter (i.e. no contradictions).

105 The selected augmentations from *AUG* introduce the following security environment aspects:

- TOE threat "Memory Access Violation, (*AUG4.T.Mem-Access*)",
- organisational security policy "Additional Specific Security Functionality, (*AUG1.P.Add-Functions*)".

106 The augmentation made in this ST introduces the following security environment aspects:

- TOE threats "Unauthorised data modification, (*T.Data-Modification*)", "Impersonating authorised users during authentication, (*T.Impersonate*)", "Cloning, (*T.Cloning*)", "DESFire code confidentiality, (*T.Confid-Applic-Code*)", "DESFire data confidentiality, (*T.Confid-Applic-Data*)", "DESFire code integrity, (*T.Integ-Applic-Code*)", "DESFire data integrity, (*T.Integ-Applic-Data*)", and "DESFire resource unavailability, (*T.Resource*)".
- organisational security policies "Confidentiality during communication, (*P.Confidentiality*)", "Transaction mechanism, (*P.Transaction*)", "Un-traceability of end-users, (*P.No-Trace*)", "Usage of hardware platform, (*P.Plat-Appl*)", and "Treatment of user data, (*P.Resp-Appl*)".
- assumptions "Usage of secure values, (*A.Secure-Values*)", and "Terminal support to ensure integrity and confidentiality, (*A.Terminal-Support*)".

107 The justification of the additional policies, additional threats, and additional assumptions provided in the next subsections shows that they do not contradict to the rationale already given in the protection profile *BSI-PP-0035* for the assumptions, policy and threats defined there.

108 In particular, the added assumptions and objectives on the environment do not contradict with the policies, threats and assumptions of the *BSI-PP-0035* Protection Profile, to which strict conformance is claimed, because they are all exclusively related to DESFire, which is out of the scope of this protection profile.

Table 6. Security Objectives versus Assumptions, Threats or Policies

Assumption, Threat or Organisational Security Policy	Security Objective	Notes
BSI.A.Plat-Appl	*BSI.OE.Plat-Appl*	Phase 1
BSI.A.Resp-Appl	*BSI.OE.Resp-Appl*	Phase 1
BSI.P.Process-TOE	*BSI.O.Identification*	Phase 2-3
BSI.A.Process-Sec-IC	*BSI.OE.Process-Sec-IC*	Phase 4-6
A.Secure-Values	*OE.Secure-Values*	Phases 5-7
A.Terminal-Support	*OE.Terminal-Support*	Phase 7
AUG1.P.Add-Functions	*AUG1.O.Add-Functions*	
P.Confidentiality	*O.Confidentiality* *OE.Terminal-Support*	
P.Transaction	*O.Transaction*	
P.No-Trace	*O.No-Trace* *O.Access-Control* *O.Authentication*	
P.Plat-Appl	*O.Plat-Appl*	
P.Resp-Appl	*O.Resp-Appl*	
BSI.T.Leak-Inherent	*BSI.O.Leak-Inherent*	
BSI.T.Phys-Probing	*BSI.O.Phys-Probing*	
BSI.T.Malfunction	*BSI.O.Malfunction*	
BSI.T.Phys-Manipulation	*BSI.O.Phys-Manipulation*	
BSI.T.Leak-Forced	*BSI.O.Leak-Forced*	
BSI.T.Abuse-Func	*BSI.O.Abuse-Func*	
BSI.T.RND	*BSI.O.RND*	
AUG4.T.Mem-Access	*AUG4.O.Mem-Access*	
T.Data-Modification	*O.Access-Control* *O.Type-Consistency* *OE.Terminal-Support*	
T.Impersonate	*O.Authentication*	
T.Cloning	*O.Access-Control* *O.Authentication*	
T.Confid-Applic-Code	*O.Firewall*	
T.Confid-Applic-Data	*O.Firewall*	
T.Integ-Applic-Code	*O.Verification* *O.Firewall*	
T.Integ-Applic-Data	*O.Firewall* *O.Shr-Res*	
T.Resource	*O.Resource*	

6.3.1 Assumption "Usage of secure values"

109 The justification related to the assumption “Usage of secure values, (*A.Secure-Values*)” is as follows:

110 Since *OE.Secure-Values* requires from the Administrator, Application Manager or the Application User to use secure values for the configuration of the authentication and access control as assumed in *A.Secure-Values*, the assumption is covered by the objective.

111 *A.Secure-Values* and *OE.Secure-Values* do not contradict with the security problem definition of the *BSI-PP-0035*, because they are only related to DESFire, which is out of the scope of this protection profile.

6.3.2 Assumption "Terminal support to ensure integrity and confidentiality"

112 The justification related to the assumption “Terminal support to ensure integrity and confidentiality, (*A.Terminal-Support*)” is as follows:

113 The objective *OE.Terminal-Support* is an immediate transformation of the assumption *A.Terminal-Support*, therefore it covers the assumption.

114 *A.Terminal-Support* and *OE.Terminal-Support* do not contradict with the security problem definition of the *BSI-PP-0035*, because they are only related to DESFire, which is out of the scope of this protection profile.

6.3.3 TOE threat "Memory Access Violation"

115 The justification related to the threat “Memory Access Violation, (*AUG4.T.Mem-Access*)” is as follows:

116 According to *AUG4.O.Mem-Access* the TOE must enforce the ***dynamic memory segmentation and protection*** so that access of software to memory areas is controlled. Any restrictions are to be defined by the ***Security IC*** Embedded Software. Thereby security violations caused by accidental or deliberate access to restricted data (which may include code) can be prevented (refer to *AUG4.T.Mem-Access*). The threat *AUG4.T.Mem-Access* is therefore removed if the objective is met.

117 The added objective for the TOE *AUG4.O.Mem-Access* does not introduce any contradiction in the security objectives for the TOE.

6.3.4 TOE threat "Unauthorised data modification"

118 The justification related to the threat “Unauthorised data modification, (*T.Data-Modification*)” is as follows:

119 According to threat *T.Data-Modification*, the TOE shall avoid that user data stored by the TOE may be modified by unauthorised subjects. The objective *O.Access-Control* requires an access control mechanism that limits the ability to modify data elements stored by the TOE. *O.Type-Consistency* ensures that data types are adhered, so that data can not be modified by abusing type-specific operations. The terminal must support this by checking the TOE responses, which is required by *OE.Terminal-Support*. Therefore *T.Data-Modification* is covered by these three objectives.

120 The added objectives for the TOE *O.Access-Control* and *O.Type-Consistency* do not introduce any contradiction in the security objectives for the TOE.

6.3.5 TOE threat "Impersonating authorised users during authentication"

121 The justification related to the threat “Impersonating authorised users during authentication, (*T.Impersonate*)” is as follows:

122 The threat is related to the fact that an unauthorised subject may try to impersonate an authorised subject during authentication, e.g. by a man-in-the middle or replay attack. The goal of *O.Authentication* is that an authentication mechanism is implemented in the TOE that prevents these attacks. Therefore the threat is covered by *O.Authentication*.

123 The added objective for the TOE *O.Authentication* does not introduce any contradiction in the security objectives for the TOE.

6.3.6 TOE threat "Cloning"

124 The justification related to the threat “Cloning, (*T.Cloning*)” is as follows:

125 The concern of *T.Cloning* is that all data stored on the TOE (including keys) may be read out in order to create a duplicate. The objective *O.Authentication* together with *O.Access-Control* requires that unauthorised users can not read any information that is restricted to the authorised subjects. The cryptographic keys used for the authentication are stored inside the TOE protected. *O.Access-Control* states that no keys used for authentication shall ever be output. Therefore the two objectives cover *T.Cloning*.

6.3.7 TOE threat "DESFire resource unavailability"

126 The justification related to the threat “DESFire resource unavailability, (*T.Resource*)” is as follows:

127 The concern of *T.Resource* is to prevent denial of service or malfunction of DESFire, that may result from an unavailability of resources. The goal of *O.Resource* is to control the availability of resources for DESFire. Therefore the threat is covered by *O.Resource*.

128 The added objective for the TOE *O.Resource* does not introduce any contradiction in the security objectives for the TOE.

6.3.8 TOE threat "DESFire code confidentiality"

129 The justification related to the threat “DESFire code confidentiality, (*T.Confid-Applic-Code*)” is as follows:

130 Since *O.Firewall* requires that the TOE ensures isolation of code between DESFire and the other applications, the code of DESFire is protected against unauthorised disclosure, therefore *T.Confid-Applic-Code* is covered by *O.Firewall*.

131 The added objective for the TOE *O.Firewall* does not introduce any contradiction in the security objectives for the TOE.

6.3.9 TOE threat "DESFire data confidentiality"

132 The justification related to the threat “DESFire data confidentiality, (*T.Confid-Applic-Data*)” is as follows:

133 Since *O.Firewall* requires that the TOE ensures isolation of data between DESFire and the other applications, the data of DESFire is protected against unauthorised disclosure, therefore *T.Confid-Applic-Data* is covered by *O.Firewall*.

6.3.10 TOE threat "DESFire code integrity"

134 The justification related to the threat “DESFire code integrity, (*T.Integ-Applic-Code*)” is as follows:

135 The threat is related to the alteration of DESFire code by an attacker. *O.Verification* requires that the TOE verifies the code integrity before its execution. Complementary, *O.Firewall* requires that the TOE ensures isolation of code between DESFire and the other applications, thus protecting the code of DESFire against unauthorised modification. Therefore the threat is covered by *O.Verification* together with *O.Firewall*.

136 The added objective for the TOE *O.Verification* does not introduce any contradiction in the security objectives for the TOE.

6.3.11 TOE threat "DESFire data integrity"

137 The justification related to the threat “DESFire data integrity, (*T.Integ-Applic-Data*)” is as follows:

138 The threat is related to the alteration of DESFire data by an attacker. Since *O.Firewall* and *O.Shr-Res* require that the TOE ensures isolation of data between DESFire and the other applications, the data of DESFire is protected against unauthorised modification, therefore *T.Integ-Applic-Data* is covered by *O.Firewall* together with *O.Shr-Res*.

139 The added objective for the TOE *O.Shr-Res* does not introduce any contradiction in the security objectives for the TOE.

6.3.12 Organisational security policy "Additional Specific Security Functionality"

140 The justification related to the organisational security policy "Additional Specific Security Functionality, (*AUG1.P.Add-Functions*)” is as follows:

141 Since *AUG1.O.Add-Functions* requires the TOE to implement exactly the same specific security functionality as required by *AUG1.P.Add-Functions*, ***and in the very same conditions,*** the organisational security policy is covered by the objective.

142 Nevertheless the security objectives *BSI.O.Leak-Inherent*, *BSI.O.Phys-Probing*, , *BSI.O.Malfunction*, *BSI.O.Phys-Manipulation* and *BSI.O.Leak-Forced* define how to implement the specific security functionality required by *AUG1.P.Add-Functions*. (Note that these objectives support that the specific security functionality is provided in a secure way as expected from *AUG1.P.Add-Functions*.) Especially *BSI.O.Leak-Inherent* and *BSI.O.Leak-Forced* refer to the protection of confidential data (User Data or TSF data) in general. User Data are also processed by the specific security functionality required by *AUG1.P.Add-Functions*.

143 The added objective for the TOE *AUG1.O.Add-Functions* does not introduce any contradiction in the security objectives for the TOE.

6.3.13 Organisational security policy "Confidentiality during communication"

144 The justification related to the organisational security policy "Confidentiality during communication, (*P.Confidentiality*)” is as follows:

145 The policy *P.Confidentiality* requires the TOE to provide the possibility to protect selected data elements from eavesdropping during contact-less communication. In addition, the data transfer is protected in a way that injected and bogus commands, within the communication

session before the protected data transfer, can be detected. The terminal must support this by checking the TOE responses, which is required by *OE.Terminal-Support*. Since *O.Confidentiality* requires that the security attribute for a data element contains an option that the communication related to this data element must be encrypted and protected, and because *OE.Terminal-Support* ensures the support by the terminal, the two objectives cover the policy.

146 The added objective for the TOE *O.Confidentiality* does not introduce any contradiction in the security objectives.

6.3.14 Organisational security policy "Transaction mechanism"

147 The justification related to the organisational security policy "Transaction mechanism, (*P.Transaction*)" is as follows:

148 According to this policy, the TOE shall be able to 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. This is exactly the goal of the objective *O.Transaction*, therefore the policy *P.Transaction* is covered by *O.Transaction*.

149 The added objective for the TOE *O.Transaction* does not introduce any contradiction in the security objectives.

6.3.15 Organisational security policy "Un-traceability of end-users"

150 The justification related to the organisational security policy "Un-traceability of end-users, (*P.No-Trace*)" is as follows:

151 The policy requires that the TOE has the ability to prevent tracing of end-users. Tracing can be performed with the UID or with any freely accessible data element stored by the TOE. The objective *O.No-Trace* 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 unauthorised subject, which includes the UID. The objectives *O.Authentication* and *O.Access-Control* provide means to authorise subjects and to implement access control to data elements in a way that unauthorised subjects can not read any element usable for tracing. Therefore the policy is covered by these three objectives.

152 The added objective for the TOE *O.No-Trace* does not introduce any contradiction in the security objectives.

6.3.16 Organisational security policy "Usage of hardware platform"

153 The justification related to the organisational security policy "Usage of hardware platform, (*P.Plat-Appl*)" is as follows:

154 The policy states that the Security IC Embedded Software included in the TOE, uses the TOE hardware according to the respective PP assumption *BSI.A.Plat-Appl*. *O.Plat-Appl* has the same objective as *BSI.OE.Plat-Appl* defined in the PP. Thus, the objective *O.Plat-Appl* covers the policy *P.Plat-Appl*.

155 The added objective for the TOE *O.Plat-Appl* does not introduce any contradiction in the security objectives.

6.3.17 Organisational security policy "Treatment of user data"

156 The justification related to the organisational security policy "Treatment of user data, (*P.Resp-Appl*)" is as follows:

157 In analogy to *P.Plat-Appl*, the policy *P.Resp-Appl* is covered in the same way by the objective *O.Resp-Appl*.

158 The added objective for the TOE *O.Resp-Appl* does not introduce any contradiction in the security objectives.

7 Security requirements

159 This chapter on security requirements contains a section on security functional requirements (SFRs) for the TOE (*Section 7.1*), a section on security assurance requirements (SARs) for the TOE (*Section 7.2*), a section on the refinements of these SARs (*Section 7.3*) as required by the "*BSI-PP-0035*" Protection Profile. This chapter includes a section with the security requirements rationale (*Section 7.4*).

7.1 Security functional requirements for the TOE

160 Security Functional Requirements (SFRs) from the "*BSI-PP-0035*" Protection Profile (PP) are drawn from *CCMB-2012-09-002*, except the following SFRs, that are **extensions** to *CCMB-2012-09-002*:

- **FCS_RNG** Generation of random numbers,
- **FMT_LIM** Limited capabilities and availability,
- **FAU_SAS** Audit data storage.

The reader can find their certified definitions in the text of the "*BSI-PP-0035*" Protection Profile.

161 All extensions to the SFRs of the "*BSI-PP-0035*" Protection Profiles (PPs) are **exclusively** drawn from *CCMB-2012-09-002*.

162 All iterations, assignments, selections, or refinements on SFRs have been performed according to section C.4 of *CCMB-2012-09-001*. They are easily identified in the following text as they appear ***as indicated here***. Note that in order to improve readability, iterations are sometimes expressed within tables.

163 In order to ease the definition and the understanding of these security functional requirements, a simplified presentation of the TOE Security Policy (TSP) is given in the following section.

164 The selected security functional requirements for the TOE, their respective origin and type are summarized in *Table 7*.

Table 7. Summary of functional security requirements for the TOE

Label	Title	Addressing	Origin	Type
FRU_FLT.2	Limited fault tolerance	Malfunction	*BSI-PP-0035*	*CCMB-2012-09-002*
FPT_FLS.1	Failure with preservation of secure state			
FMT_LIM.1	Limited capabilities	Abuse of TEST functionality	*BSI-PP-0035*	Extended
FMT_LIM.2	Limited availability			
FAU_SAS.1	Audit storage	Lack of TOE identification	*BSI-PP-0035* Operated	

Table 7. Summary of functional security requirements for the TOE (continued)

Label	Title	Addressing	Origin	Type
FPT_PHP.3	Resistance to physical attack	Physical manipulation & probing	*BSI-PP-0035*	*CCMB-2012-09-002*
FDP_ITT.1	Basic internal transfer protection	Leakage		
FPT_ITT.1	Basic internal TSF data transfer protection			
FDP_IFC.1	Subset information flow control			
FCS_RNG.1	Random number generation	Weak cryptographic quality of random numbers	*BSI-PP-0035* Operated	Extended
FCS_COP.1	Cryptographic operation	Cipher scheme support	*AUG* #1 Operated	*CCMB-2012-09-002*
FCS_CKM.1 (if Neslib is embedded only)	Cryptographic key generation		Security Target Operated	
FDP_ACC.2 [Memories]	Complete access control	Memory access violation	Security Target Operated	
FDP_ACF.1 [Memories]	Security attribute based access control		*AUG* #4 Operated	
FMT_MSA.3 [Memories]	Static attribute initialisation	Correct operation		
FMT_MSA.1 [Memories]	Management of security attribute			
FMT_SMF.1 [Memories]	Specification of management functions		Security Target Operated	
FMT_SMR.1 [MIFARE]	Security roles	DESFire access control (if DESFire is embedded only)	Security Target Operated	*CCMB-2012-09-002*
FDP_ACC.1 [MIFARE]	Subset access control			
FDP_ACF.1 [MIFARE]	Security attribute based access control			
FMT_MSA.3 [MIFARE]	Static attribute initialisation			
FMT_MSA.1 [MIFARE]	Management of security attribute			
FMT_SMF.1 [MIFARE]	Specification of management functions			
FDP_ITC.2 [MIFARE]	Import of user data with security attributes			
FPT_TDC.1 [MIFARE]	Inter-TSF basic TSF data consistency			

Table 7. Summary of functional security requirements for the TOE (continued)

Label	Title	Addressing	Origin	Type
FIA_UID.2 [MIFARE]	User identification before any action	DESFire confidentiality and authentication (if DESFire is embedded only)	Security Target Operated	CCMB-2012-09-002
FIA_UAU.2 [MIFARE]	User authentication before any action			
FIA_UAU.5 [MIFARE]	Multiple authentication mechanisms			
FMT_MTD.1 [MIFARE]	Management of TSF data			
FPT_TRP.1 [MIFARE]	Trusted path			
FCS_CKM.4 [MIFARE]	Cryptographic key destruction			
FDP_ROL.1 [MIFARE]	Basic rollback	DESFire robustness (if DESFire is embedded only)		
FPT_RPL.1 [MIFARE]	Replay detection			
FPR_UNL.1 [MIFARE]	Unlinkability			
FPT_TST.1 [MIFARE]	TSF testing	DESFire correct operation (if DESFire is embedded only)		
FRU_RSA.2 [MIFARE]	Minimum and maximum quotas			
FDP_RIP.1 [MIFARE]	Subset residual information protection	DESFire intrinsic confidentiality and integrity (if DESFire is embedded only)		
FDP_ACC.1 [MIFARE_FWL]	Subset access control			
FDP_ACF.1 [MIFARE_FWL]	Security attribute based access control			
FMT_MSA.3 [MIFARE_FWL]	Static attribute initialisation			

7.1.1 Security Functional Requirements from the Protection Profile

Limited fault tolerance (FRU_FLT.2)

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

Failure with preservation of secure state (FPT_FLS.1)

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

167 Refinement:

The term “failure” above also covers “circumstances”. The TOE prevents failures for the “circumstances” defined above.

Regarding application note 15 of *BSI-PP-0035*, the TOE provides information on the operating conditions monitored during Security IC Embedded Software execution and after a warm reset. No audit requirement is however selected in this Security Target.

Limited capabilities (FMT_LIM.1)

168 The TSF shall be designed and implemented in a manner that limits their capabilities so that in conjunction with “Limited availability (FMT_LIM.2)” the following policy is enforced: Limited capability and availability Policy.

Limited availability (FMT_LIM.2)

169 The TSF shall be designed and implemented in a manner that limits their availability so that in conjunction with “Limited capabilities (FMT_LIM.1)” the following policy is enforced: Limited capability and availability Policy.

170 *SFP_1: Limited capability and availability Policy*

Deploying Test Features after TOE Delivery does not allow User Data to be disclosed or manipulated, TSF data to be disclosed or manipulated, software to be reconstructed and no substantial information about construction of TSF to be gathered which may enable other attacks.

Audit storage (FAU_SAS.1)

171 The TSF shall provide ***the test process before TOE Delivery*** with the capability to store the ***Initialisation Data and/or Pre-personalisation Data and/or supplements of the Security IC Embedded Software*** in the ***NVM***.

Resistance to physical attack (FPT_PHP.3)

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

173 Refinement:

The TSF will implement appropriate mechanisms to continuously counter physical manipulation and physical probing. Due to the nature of these attacks (especially manipulation) the TSF can by no means detect attacks on all of its elements. Therefore, permanent protection against these attacks is required ensuring that security functional requirements are enforced. Hence, “automatic response” means here (i)assuming that there might be an attack at any time and (ii)countermeasures are provided at any time.

Basic internal transfer protection (FDP_ITT.1)

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

Basic internal TSF data transfer protection (FPT_ITT.1)

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

176 Refinement:

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

This requirement is equivalent to FDP_ITT.1 above but refers to TSF data instead of User Data. Therefore, it should be understood as to refer to the same ***Data Processing Policy*** defined under FDP_IFC.1 below.

Subset information flow control (FDP_IFC.1)

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

178 *SFP_2: Data Processing Policy*

User Data and TSF data shall not be accessible from the TOE except when the Security IC Embedded Software decides to communicate the User Data via an external interface. The protection shall be applied to confidential data only but without the distinction of attributes controlled by the Security IC Embedded Software.

Random number generation (FCS_RNG.1)

179 The TSF shall provide a ***physical*** random number generator that implements a ***total failure test of the random source.***

180 The TSF shall provide random numbers that meet ***P2 class of*** *BSI-AIS31*.

7.1.2 Additional Security Functional Requirements for the cryptographic services

Cryptographic operation (FCS_COP.1)

181 The TSF shall perform ***the operations in*** *Table 8* in accordance with a specified cryptographic algorithm ***in*** *Table 8* and cryptographic key sizes ***of*** *Table 8* that meet the ***standards in*** *Table 8****. The list of operations depends on the presence of Neslib, as indicated in*** *Table 8* ***(Restrict).***

Table 8. FCS_COP.1 iterations (cryptographic operations)

Restrict	Iteration label	[assignment: list of cryptographic operations]	[assignment: cryptographic algorithm]	[assignment: cryptographic key sizes]	[assignment: list of standards]
Even without Neslib	EDES	* encryption * decryption - in Cipher Block Chaining (CBC) mode - in Electronic Code Book (ECB) mode * MAC computation in CBC-MAC	Data Encryption Standard (DES)	56 bits	*FIPS PUB 46-3* *ISO/IEC 9797-1* *ISO/IEC 10116*
			Triple Data Encryption Standard (3DES)	168 bits	

Table 8. FCS_COP.1 iterations (cryptographic operations) (continued)

Restrict	Iteration label	[assignment: list of cryptographic operations]	[assignment: cryptographic algorithm]	[assignment: cryptographic key sizes]	[assignment: list of standards]
Even without Neslib	AES	* encryption (cipher) * decryption (inverse cipher) * key expansion * randomize	Advanced Encryption Standard	128, 192 and 256 bits	*FIPS PUB 197*
If Neslib	RSA	* RSA public key operation * RSA private key operation without the Chinese Remainder Theorem * RSA private key operation with the Chinese Remainder Theorem	Rivest, Shamir & Adleman's	up to 4096 bits	*PKCS #1 V2.1*
If Neslib	ECC	* private scalar multiplication * prepare Jacobian * public scalar multiplication * point validity check * convert Jacobian to affine coordinates * general point addition * point expansion * point compression	Elliptic Curves Cryptography on GF(p)	up to 640 bits	*IEEE 1363-2000, chapter 7* *IEEE 1363a-2004*
If Neslib	SHA	* SHA-1 * SHA-224 * SHA-256 * SHA-384 * SHA-512 * Protected SHA-1	Secure Hash Algorithm	assignment pointless because algorithm has no key	*FIPS PUB 180-1* *FIPS PUB 180-2* *ISO/IEC 10118-3:1998*

Cryptographic key generation (FCS_CKM.1)

182 If Neslib is embedded only, the TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm***, in*** *Table 9****,*** and specified cryptographic key sizes ***of*** *Table 9* that meet the following ***standards in*** *Table 9****.***

Table 9. FCS_CKM.1 iterations (cryptographic key generation)

Iteration label	[assignment: cryptographic key generation algorithm]	[assignment: cryptographic key sizes]	[assignment: list of standards]
Prime generation	prime generation and RSA prime generation algorithm	up to 2048 bits	*FIPS PUB 140-2* *FIPS PUB 186*
Protected prime generation	prime generation and RSA prime generation algorithm, protected against side channel attacks	up to 2048 bits	*FIPS PUB 140-2* *FIPS PUB 186*
RSA key generation	RSA key pair generation algorithm	up to 4096 bits	*FIPS PUB 140-2* *ISO/IEC 9796-2* *PKCS #1 V2.1*
Protected RSA key generation	RSA key pair generation algorithm, protected against side channel attacks	up to 4096 bits	*FIPS PUB 140-2* *ISO/IEC 9796-2* *PKCS #1 V2.1*

7.1.3 Additional Security Functional Requirements for the memories protection

Static attribute initialisation (FMT_MSA.3) [Memories]

183 The TSF shall enforce the ***Dynamic Memory Access Control Policy*** to provide ***minimally protective***[(a)] default values for security attributes that are used to enforce the SFP.

184 The TSF shall allow ***none*** to specify alternative initial values to override the default values when an object or information is created.

Application note:
The security attributes are the set of access rights currently defined. They are dynamically attached to the subjects and objects locations, i.e. each logical address.

Management of security attributes (FMT_MSA.1) [Memories]

185 The TSF shall enforce the ***Dynamic Memory Access Control Policy*** to restrict the ability to ***modify*** the security attributes ***current set of access rights*** to ***software running in privileged mode.***

Complete access control (FDP_ACC.2) [Memories]

186 The TSF shall enforce the ***Dynamic Memory Access Control Policy*** on ***all subjects (software), all objects (data including code stored in memories)*** and all operations among subjects and objects covered by the SFP.

187 The TSF shall ensure that all operations between any subject controlled by the TSF and any object controlled by the TSF are covered by an access control SFP.

a. See the Datasheet referenced in *Section 9* for actual values.

Security attribute based access control (FDP_ACF.1) [Memories]

188 The TSF shall enforce the ***Dynamic Memory Access Control Policy*** to objects based on the following: ***software mode, the object location, the operation to be performed, and the current set of access rights.***

189 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: ***the operation is allowed if and only if the software mode, the object location and the operation matches an entry in the current set of access rights.***

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

191 The TSF shall explicitly deny access of subjects to objects based on the following additional rules: ***in User configuration, any access (read, write, execute) to the OST ROM is denied, and in User configuration, any write access to the ST NVM is denied.***

Note: It should be noted that this level of policy detail is not needed at the application level. The composite Security Target writer should describe the ES access control and information flow control policies instead. Within the ES High Level Design description, the chosen setting of IC security attributes would be shown to implement the described policies relying on the IC SFP presented here.

192 The following SFP ***Dynamic Memory Access Control Policy*** is defined for the requirement "Security attribute based access control (FDP_ACF.1)":

193 SFP_3: Dynamic Memory Access Control Policy

194 The TSF must control read, write, execute accesses of software to data, based on the software mode and on the current set of access rights.

Specification of management functions (FMT_SMF.1) [Memories]

195 The TSF will be able to perform the following management functions: ***modification of the current set of access rights security attributes by software running in privileged mode, supporting the Dynamic Memory Access Control Policy.***

7.1.4 Additional Security Functional Requirements related to DESFire

196 The following SFRs are extensions to "*BSI-PP-0035*" Protection Profile (PP), related to the capabilities and protections of DESFire.

197 They are only valid in case DESFire is embedded.

198 **Note**: MIFARE DESFire EV1 library directly relies upon the following IC SFRs:

- FRU_FLT.2 in providing services as part of the security countermeasures implemented in the library,
- FPT_FLS.1 in order to generate a software reset,
- FCS_RNG.1 for the provision of random numbers,
- FCS_COP.1 [EDES] for DES cryptographic operations,
- FCS_COP.1 [AES] for AES cryptographic operations.

199 It also relies upon the other SFRs (except those of Neslib), which provide general low level security mechanisms.

Security roles (FMT_SMR.1) [MIFARE]

200 The TSF shall maintain the roles ***Administrator, Application Manager, Application User and Everybody***.

201 The TSF shall be able to associate users with roles.

202 ***Note: Based on the definition, Nobody is not considered as a role.***

Subset access control (FDP_ACC.1) [MIFARE]

203 The TSF shall enforce the ***MIFARE Access Control Policy*** on ***all subjects, objects, operations and attributes defined by the MIFARE Access Control Policy.***

Security attribute based access control (FDP_ACF.1) [MIFARE]

204 The TSF shall enforce the ***MIFARE Access Control Policy*** to objects based on the following: ***all subjects, objects and attributes***.

205 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:

- ***The Administrator can create and delete applications.***
- ***The Application Manager of an application can delete this application, create data file and values within this application, delete data files and values within this application.***
- ***An Application User can read or write a data file; read, increase or decrease a value based on the access control settings in the respective file attribute.***

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

- ***Everybody can create applications if this is allowed by a specific card attribute.***
- ***Everybody can create and delete data files or values of a specific application if this is allowed by a specific application attribute.***
- ***Everybody can read or write a data file; read, increase or decrease a value if this is allowed by a specific file attribute.***

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

- ***Nobody can read or write a data file; read, increase or decrease a value if this is explicitly set for the respective operation on the respective data file or value.***

208 The following SFP ***MIFARE Access Control Policy*** is defined for the requirement "Security attribute based access control (FDP_ACF.1) [MIFARE]":

209 *SFP_4: MIFARE Access Control Policy*

210 *The Security Function Policy (SFP) MIFARE Access Control Policy uses the following definitions:*

211 *The subjects are:*

- *The Administrator i.e. the subject that owns or has access to the card master key.*
- *The Application Manager i.e. the subject that owns or has access to an application master key. Note that the TOE supports multiple applications and therefore multiple*

Application Managers, however for one application there is only one Application Manager.

- *The Application User i.e. the subject that owns or has access to a key that allows to perform operations with application objects. Note that the TOE supports multiple Application Users within each application and the assigned rights to the Application Users can be different, which allows to have more or less powerful Application Users.*
- *Any other subject belongs to the role Everybody. This includes the card holder (i.e. end-user) and any other subject e.g. an attacker. These subjects do not possess any key and can not perform operations that are restricted to the Administrator, Application Manager and Application User.*
- *The term Nobody will be used to explicitly indicate that no rights are granted to any subject.*

212 *The objects are:*

- *The Card itself.*
- *The card can store a number of Applications.*
- *An application can store a number of Data Files of different types.*
- *One specific type of data file are Values.*

213 *Note that data files and values can be grouped in standard files and backup files, with values belonging to the group of backup files. When the term "file" is used without further information then both data files and values are meant.*

214 *The operations that can be performed with the objects are:*

- *read a value or data from a data file,*
- *write data to a data file,*
- *increase a value (with a limit or unlimited),*
- *decrease a value,*
- *create an application, a value or a data file,*
- *delete an application, a value or a data file and*
- *modify attribute of the card, an application, a value or a data file. Note that 'freeze' will be used as specific form of modification that prev*ents any further modify.

215 *The security attributes are:*

- *Attributes of the card, applications, values and data files.*
  There is a set of attributes for the card, a set of attributes for every application and a set of attributes for every single file within an application.
  The term "card attributes" will be used for the set of attributes related to the card, the term "application attributes" will be used for the set of application attributes and the term "file attributes" will be used for the attributes of values and data files.

216 *Note that subjects are authorised by cryptographic keys. These keys are considered as authentication data and not as security attributes. The card has a card master key. Every application has an application master key and a variable number of keys used for operations on data files or values (all these keys are called application keys). The application keys within an application are numbered.*

217 *Implications of the MIFARE Access Control Policy:*

218 *The MIFARE 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 (Administrator, Application Manager, Application User), but regarded as 'Everybody' 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 Administrator can have the exclusive right to create and delete applications on the Smart Card, however he can also grant this privilege to Everybody. Additionally, changing the Smart Card attributes is reserved for the Administrator. Application keys, at delivery time should be personalized to a preliminary, temporary key only known to the Administrator and the Application Manager.*
- *At application personalization time, the Application Manager uses the preliminary application key in order to personalize the application keys, whereas all keys, except the application master key, can be personalized to a preliminary, temporary key only known to the Application Manager and the Application User. Furthermore, the Application Manager has the right to create files within his application scope.*

Static attribute initialisation (FMT_MSA.3) [MIFARE]

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

220 The TSF shall allow ***no subject*** to specify alternative initial values to override the default values when an object or information is created.

221 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.

Management of security attributes (FMT_MSA.1) [MIFARE]

222 The TSF shall enforce the ***MIFARE Access Control Policy*** to restrict the ability to ***modify or freeze*** the security attributes ***card attributes, application attributes and file attributes*** to the ***Administrator, Application Manager and Application User, respectively***.

223 Refinement:

The detailed management abilities are:

- The Administrator can modify the card attributes. The card attributes contain a flag that when set will prevent any further change of the card attributes, thereby allowing to freeze the card attributes.
- The Application Manager can modify the application attributes. The application attributes contain a flag that when set will prevent any further change of the application attributes, thereby allowing to freeze the application attributes.
- The Application Manager can decide to restrict the ability to modify the file attributes to the Application Manager, an Application User, Everybody or to Nobody. The restriction to Nobody is equivalent to freezing the file attributes.
- As an implication of the last rule, any subject that receives the modify abilities from the Application Manger gets these abilities transferred.
- The implication given in the previous rule includes the possibility for an Application User to modify the file attributes if the Application Manager decides to transfer this

ability. If there is no such explicit transfer an Application User does not have the ability to modify the file attributes.

Specification of Management Functions (FMT_SMF.1) [MIFARE]

224 The TSF shall be capable of performing the following security management functions:

- ***Authenticating a user,***
- ***Invalidating the current authentication state based on the functions: Selecting an application or the card, Changing a key, Occurrence of any error during the execution of a command, Reset,***
- ***Changing a security attribute,***
- ***Creating or deleting an application, a value or a data file.***

Import of user data with security attributes (FDP_ITC.2) [MIFARE]

225 The TSF shall enforce the ***MIFARE Access Control Policy*** when importing user data, controlled under the SFP, from outside of the TOE.

226 The TSF shall use the security attributes associated with the imported user data.

227 The TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the user data received.

228 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.

229 The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: ***no additional rules***.

Inter-TSF basic TSF data consistency (FPT_TDC.1) [MIFARE]

230 The TSF shall provide the capability to consistently interpret ***data files and values*** when shared between the TSF and another trusted IT product.

231 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*** when interpreting the TSF data from another trusted IT product.

Application note:
The TOE does not interpret the contents of the data, e.g. it can not determine if data stored in a specific data file is an identification number that adheres to a specific format. Instead the TOE distinguishes different types of files and ensures that type-specific boundaries can not be violated, e.g. values do not overflow, single records are limited by their size and cyclic records are handled correctly.

Cryptographic key destruction (FCS_CKM.4) [MIFARE]

232 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method ***overwriting of memory*** that meets the following: ***none***.

User identification before any action (FIA_UID.2) [MIFARE]

233 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’.

User authentication before any action (FIA_UAU.2) [MIFARE]

234 The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

Multiple authentication mechanisms (FIA_UAU.5) [MIFARE]

235 The TSF shall provide ***‘none’ and cryptographic authentication*** to support user authentication.

236 The TSF shall authenticate any user's claimed identity according to the ***following rules:***

- ***The ‘none’ authentication is performed with anyone who communicates with the TOE without issuing an explicit authentication request. The ‘none’ authentication implicitly and solely authorises the ‘Everybody’ subject.***
- ***The cryptographic authentication is used to authorise the Administrator, Application Manager and Application User.***

Management of TSF data (FMT_MTD.1) [MIFARE]

237 The TSF shall restrict the ability to ***change_default, modify or freeze*** the ***card master key, application master keys and application keys*** to ***the Administrator, Application Manager and Application User***.

238 Refinement:

The detailed management abilities are:

- The Administrator can modify the card master key. The card attributes contain a flag that when set will prevent any further change of the card master key, thereby allowing to freeze the card master key.
- The Administrator can change the default key that is used for the application master key and for the application keys when an application is created.
- The Application Manager of an application can modify the application master key of this application. The application attributes contain a flag that when set will prevent any further change of the application master key, thereby allowing to freeze the application master key.
- The Application Manager can decide to restrict the ability to modify the application keys to the Application Manager, the Application Users or to Nobody. The restriction to Nobody is equivalent to freezing the application keys. The Application Users can either change their own keys or one Application User can be defined that can change all keys of the Application Users within an application.
- As an implication of the last rule, any subject that receives the modify abilities from the Application Manager gets these abilities transferred.

Trusted path (FTP_TRP.1) [MIFARE]

239 The TSF shall provide a communication path between itself and ***remote*** users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from ***modification or disclosure***.

240 The TSF shall permit ***remote users*** to initiate communication via the trusted path.

241 The TSF shall require the use of the trusted path for ***authentication requests with DES and AES, confidentiality and/or data integrity verification for data transfers protected with AES and based on a setting in the file attributes***.

Basic rollback (FDP_ROL.1) [MIFARE]

242 The TSF shall enforce ***the MIFARE Access Control Policy*** to permit the rollback of the ***operations that modify the value or data file objects*** on the ***backup files***.

243 The TSF shall permit operations to be rolled back within the ***scope of the current transaction, which is defined by the following limitative events: chip reset, (re-) authentication (either successful or not), select command, explicit commit, explicit abort, command failure***.

Replay detection (FPT_RPL.1) [MIFARE]

244 The TSF shall detect replay for the following entities: ***authentication requests with DES and AES, confidentiality and/or data integrity verification for data transfers protected with AES and based on a setting in the file attributes***.

245 The TSF shall perform ***rejection of the request*** when replay is detected.

Unlinkability (FPR_UNL.1) [MIFARE]

246 The TSF shall ensure that ***unauthorised subjects other than the card holder*** are unable to determine whether ***any operation of the TOE were caused by the same user***.

TSF testing (FPT_TST.1) [MIFARE]

247 The TSF shall run a suite of self tests ***during initial start-up and periodically during normal operation*** to demonstrate the correct operation of ***DESFire***.

248 The TSF shall provide authorised users with the capability to verify the integrity of ***the DESFire code***.

249 The TSF shall provide authorised users with the capability to verify the integrity of ***DESFire***.

Application note:
DESFire itself is the authorised user that verifies the integrity of its own code and execution.

Minimum and maximum quotas (FRU_RSA.2) [MIFARE]

250 The TSF shall enforce maximum quotas of the following resources ***NVM and RAM*** that ***subjects*** can use ***simultaneously***.

251 The TSF shall ensure the provision of minimum quantity of ***the NVM and the RAM*** that is available for ***subjects*** to use ***simultaneously***.

Application note:
The subjects addressed here are DESFire, and all other applications running on the TOE. The goal is to ensure that DESFire always have enough NVM and RAM for its own usage.

Subset residual information protection (FDP_RIP.1) [MIFARE]

252 The TSF shall ensure that any previous information content of a resource is made unavailable upon the ***deallocation of the resource from*** the following objects: ***DESFire***.

Subset access control (FDP_ACC.1) [MIFARE_FWL]

253 The TSF shall enforce the ***MIFARE Firewall Access Control Policy*** on ***the DESFire code and data.***

Security attribute based access control (FDP_ACF.1) [MIFARE_FWL]

254 The TSF shall enforce the ***MIFARE Firewall Access Control Policy*** to objects based on the following: ***DESFire code and data***.

255 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: ***An application cannot read, write, compare any piece of data or code belonging to DESFire.***

256 The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: ***None***.

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

- ***An application cannot read, write, compare any piece of data or code belonging to DESFire.***

258 The following SFP ***MIFARE Firewall Access Control Policy*** is defined for the requirement "Security attribute based access control (FDP_ACF.1) [MIFARE_FWL]":

259 *SFP_5: MIFARE Firewall Access Control Policy*

260 *An application cannot read, write, compare any piece of data or code belonging to DESFire.*

Static attribute initialisation (FMT_MSA.3) [MIFARE_FWL]

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

262 The TSF shall allow ***no subject*** to specify alternative initial values to override the default values when an object or information is created.

7.2 TOE security assurance requirements

263 Security Assurance Requirements for the TOE for the evaluation of the TOE are those taken from the Evaluation Assurance Level ***5*** (EAL***5***) and augmented by taking the following components:

- ALC_DVS.2 and AVA_VAN.5.

264 Regarding application note 21 of *BSI-PP-0035*, the continuously increasing maturity level of evaluations of Security ICs justifies the selection of a higher-level assurance package.

265 The set of security assurance requirements (SARs) is presented in *Table 10*, indicating the origin of the requirement.

Table 10. TOE security assurance requirements

Label	Title	Origin
ADV_ARC.1	Security architecture description	EAL5/*BSI-PP-0035*
ADV_FSP.5	Complete semi-formal functional specification with additional error information	EAL5

Table 10. TOE security assurance requirements (continued)

Label	Title	Origin
ADV_IMP.1	Implementation representation of the TSF	EAL5/*BSI-PP-0035*
ADV_INT.2	Well-stuctured internals	EAL5
ADV_TDS.4	Semiformal modular design	EAL5
AGD_OPE.1	Operational user guidance	EAL5/*BSI-PP-0035*
AGD_PRE.1	Preparative procedures	EAL5/*BSI-PP-0035*
ALC_CMC.4	Production support, acceptance procedures and automation	EAL5/*BSI-PP-0035*
ALC_CMS.5	Development tools CM coverage	EAL5
ALC_DEL.1	Delivery procedures	EAL5/*BSI-PP-0035*
ALC_DVS.2	Sufficiency of security measures	*BSI-PP-0035*
ALC_LCD.1	Developer defined life-cycle model	EAL5/*BSI-PP-0035*
ALC_TAT.2	Compliance with implementation standards	EAL5
ATE_COV.2	Analysis of coverage	EAL5/*BSI-PP-0035*
ATE_DPT.3	Testing: modular design	EAL5
ATE_FUN.1	Functional testing	EAL5/*BSI-PP-0035*
ATE_IND.2	Independent testing - sample	EAL5/*BSI-PP-0035*
AVA_VAN.5	Advanced methodical vulnerability analysis	*BSI-PP-0035*

7.3 Refinement of the security assurance requirements

266 As *BSI-PP-0035* defines refinements for selected SARs, these refinements are also claimed in this Security Target.

267 The main customizing is that the IC Dedicated Software is an operational part of the TOE after delivery, although it is not available to the user.

268 Regarding application note 22 of *BSI-PP-0035*, the refinements for all the assurance families have been reviewed for the hierarchically higher-level assurance components selected in this Security Target.

269 The text of the impacted refinements of *BSI-PP-0035* is reproduced in the next sections.

270 For reader's ease, an impact summary is provided in *Table 11*.

Table 11. Impact of EAL5 selection on *BSI-PP-0035* refinements

Assurance Family	*BSI-PP-0035* Level	ST Level	Impact on refinement
ADO_DEL	1	1	None
ALC_DVS	2	2	None
ALC_CMS	4	5	None, refinement is still valid
ALC_CMC	4	4	None

Table 11. Impact of EAL5 selection on *BSI-PP-0035* refinements (continued)

Assurance Family	*BSI-PP-0035* Level	ST Level	Impact on refinement
ADV_ARC	1	1	None
ADV_FSP	4	5	Presentation style changes, IC Dedicated Software is included
ADV_IMP	1	1	None
ATE_COV	2	2	IC Dedicated Software is included
AGD_OPE	1	1	None
AGD_PRE	1	1	None
AVA_VAN	5	5	None

7.3.1 Refinement regarding functional specification (ADV_FSP)

271 ~~Although the IC Dedicated Test Software is a part of the TOE, the test functions of the IC Dedicated Test Software are not described in the Functional Specification because the IC Dedicated Test Software is considered as a test tool delivered with the TOE but not providing security functions for the operational phase of the TOE.~~ ***The IC Dedicated Software provides security functionalities as soon as the TOE becomes operational (boot software). These are properly identified in the delivered documentation.***

272 The Functional Specification ***refers to datasheet to*** trace security features that do not provide any external interface but that contribute to fulfil the SFRs e.g. like physical protection. Thereby they are part of the complete instantiation of the SFRs.

273 The Functional Specification ***refers to design specifications to detail the*** mechanisms against physical attacks ***described*** in a more general way only, but detailed enough to be able to support Test Coverage Analysis also for those mechanisms where inspection of the layout is of relevance or tests beside the TSFI may be needed.

274 The Functional Specification ***refers to data sheet to*** specify operating conditions of the TOE. These conditions include but are not limited to the frequency of the clock, the power supply, and the temperature.

275 All functions and mechanisms which control access to the functions provided by the IC Dedicated Test Software (refer to the security functional requirement (FMT_LIM.2)) ***are part of the*** Functional Specification. Details will be given in the document for ADV_ARC~~, refer to Section 6.2.1.5.~~ In addition, all these functions and mechanisms ***are*** subsequently ~~be~~ refined according to all relevant requirements of the Common Criteria assurance class ADV because these functions and mechanisms are active after TOE Delivery and need to be part of the assurance aspects Tests (class ATE) and Vulnerability Assessment (class AVA). Therefore, all necessary information ***is*** provided to allow tests and vulnerability assessment.

276 Since the selected higher-level assurance component requires a security functional specification presented in a “semi-formal style" (ADV_FSP.5.2C) the changes affect the style of description, the *BSI-PP-0035* refinements can be applied with changes covering the IC Dedicated Test Software and are valid for ADV_FSP.5.

7.3.2 Refinement regarding test coverage (ATE_COV)

277 The TOE ***is*** tested under different operating conditions within the specified ranges. These conditions include but are not limited to the frequency of the clock, the power supply, and the temperature. This means that “Fault tolerance (FRU_FLT.2)” ***is*** proven for the complete TSF. The tests ~~must~~ also cover functions which may be affected by “ageing” (such as EEPROM writing).

278 The existence and effectiveness of measures against physical attacks (as specified by the functional requirement FPT_PHP.3) cannot be tested in a straightforward way. Instead ***STMicroelectronics provides*** evidence that the TOE actually has the particular physical characteristics (especially layout design principles). This ***is*** done by checking the layout (implementation or actual) in an appropriate way. The required evidence pertains to the existence of mechanisms against physical attacks (unless being obvious).

279 ~~The IC Dedicated Test Software is seen as a “test tool” being delivered as part of the TOE. However, the Test Features do not provide security functionality. Therefore, Test Features need not to be covered by the Test Coverage Analysis but all functions and mechanisms which limit the capability of the functions (cf. FMT_LIM.1) and control access to the functions (cf. FMT_LIM.2) provided by the IC Dedicated Test Software must be part of the Test Coverage Analysis.~~ ***The IC Dedicated Software provides security functionalities as soon as the TOE becomes operational (boot software). These are part of the Test Coverage Analysis.***

7.4 Security Requirements rationale

7.4.1 Rationale for the Security Functional Requirements

280 Just as for the security objectives rationale of *Section 6.3*, the main line of this rationale is that the inclusion of all the security requirements of the *BSI-PP-0035* protection profile, together with those in *AUG*, and with those introduced in this Security Target, guarantees that all the security objectives identified in *Section 6* are suitably addressed by the security requirements stated in this chapter, and that the latter together form an internally consistent whole.

281 As origins of security objectives have been carefully kept in their labelling, and origins of security requirements have been carefully identified in *Table 7*, it can be verified that the justifications provided by the *BSI-PP-0035* protection profile and *AUG* can just be carried forward to their union.

282 From *Table 5*, it is straightforward to identify two additional security objectives for the TOE (*AUG1.O.Add-Functions* and *AUG4.O.Mem-Access*) tracing back to *AUG*, and twelve additional objectives (*O.Access-Control*, *O.Authentication*, *O.Confidentiality*, *O.Type-Consistency*, *O.Transaction*, *O.No-Trace*, *O.Plat-Appl*, *O.Resp-Appl*, *O.Resource*, *O.Verification*, *O.Firewall* and *O.Shr-Res*) introduced in this Security Target. This rationale must show that security requirements suitably address these fourteen.

283 Furthermore, a more careful observation of the requirements listed in *Table 7* shows that:

- there are security requirements introduced from *AUG* (*FCS_COP.1*, *FDP_ACC.2 [Memories]*, *FDP_ACF.1 [Memories]*, *FMT_MSA.3 [Memories]* and *FMT_MSA.1 [Memories]*),
- there are additional security requirements introduced by this Security Target (*FCS_CKM.1*, *FMT_SMF.1 [Memories]*, *FMT_SMR.1 [MIFARE]*, *FDP_ACC.1*

[MIFARE], *FDP_ACF.1 [MIFARE]*, *FMT_MSA.3 [MIFARE]*, *FMT_MSA.1 [MIFARE]*, *FMT_SMF.1 [MIFARE]*, *FDP_ITC.2 [MIFARE]*, *FPT_TDC.1 [MIFARE]*, *FIA_UID.2 [MIFARE]*, *FIA_UAU.2 [MIFARE]*, *FIA_UAU.5 [MIFARE]*, *FMT_MTD.1 [MIFARE]*, *FPT_TRP.1 [MIFARE]*, *FCS_CKM.4 [MIFARE]*, *FDP_ROL.1 [MIFARE]*, *FPT_RPL.1 [MIFARE]*, *FPR_UNL.1 [MIFARE]*, *FPT_TST.1 [MIFARE]*, *FRU_RSA.2 [MIFARE]*, *FDP_RIP.1 [MIFARE]*, *FDP_ACC.1 [MIFARE_FWL]*, *FDP_ACF.1 [MIFARE_FWL]*, and *FMT_MSA.3 [MIFARE_FWL]*, and various assurance requirements of EAL5).

284 Though it remains to show that:

- security objectives from this Security Target and from *AUG* are addressed by security requirements stated in this chapter,
- additional security requirements from this Security Target and from *AUG* are mutually supportive with the security requirements from the *BSI-PP-0035* protection profile, and they do not introduce internal contradictions,
- all dependencies are still satisfied.

285 The justification that the additional security objectives are suitably addressed, that the additional security requirements are mutually supportive and that, together with those already in *BSI-PP-0035*, they form an internally consistent whole, is provided in the next subsections.

7.4.2 Additional security objectives are suitably addressed

Security objective “Dynamic Area based Memory Access Control (*AUG4.O.Mem-Access*)”

286 The justification related to the security objective “***Dynamic*** Area based Memory Access Control (*AUG4.O.Mem-Access*)” is as follows:

287 The security functional requirements "*Complete access control (FDP_ACC.2) [Memories]*" ***and*** "*Security attribute based access control (FDP_ACF.1) [Memories]*", with the related Security Function Policy (SFP) “***Dynamic Memory Access Control Policy***” exactly require to implement a ***Dynamic*** area based memory access control as demanded by *AUG4.O.Mem-Access*. Therefore, *FDP_ACC.2 [Memories]* ***and*** *FDP_ACF.1 [Memories]* with ***their*** SFP ***are*** suitable to meet the security objective.

288 The security functional requirement "*Static attribute initialisation (FMT_MSA.3) [Memories]*" requires that the TOE provides default values for security attributes. The ability to update the security attributes is restricted to privileged subject(s) ***as further detailed in the security functional requirement*** "*Management of security attributes (FMT_MSA.1) [Memories]*". These management functions ensure that the required access control can be realised using the functions provided by the TOE.

Security objective “Additional Specific Security Functionality (*AUG1.O.Add-Functions*)”

289 The justification related to the security objective “Additional Specific Security Functionality (*AUG1.O.Add-Functions*)” is as follows:

290 The security functional requirements “*Cryptographic operation (FCS_COP.1)*” ***and*** "*Cryptographic key generation (FCS_CKM.1)*" exactly require those functions to be implemented that are demanded by *AUG1.O.Add-Functions*. Therefore, *FCS_COP.1* is suitable to meet the security objective, ***together with*** *FCS_CKM.1*.

Security objective “Access control for DESFire (*O.Access-Control*)”

291 The justification related to the security objective “Access control for DESFire (*O.Access-Control*)” is as follows:

292 The security functional requirement "*Security roles (FMT_SMR.1) [MIFARE]*" defines the roles of the MIFARE Access Control Policy.
The security functional requirements "*Subset access control (FDP_ACC.1) [MIFARE]*" and "*Security attribute based access control (FDP_ACF.1) [MIFARE]*" define the rules and "*Static attribute initialisation (FMT_MSA.3) [MIFARE]*" and "*Management of security attributes (FMT_MSA.1) [MIFARE]*" the attributes that the access control is based on.
The security functional requirement "*Management of TSF data (FMT_MTD.1) [MIFARE]*" provides the rules for the management of the authentication data.
The management functions are defined by "*Specification of Management Functions (FMT_SMF.1) [MIFARE]*".
Since the TOE stores data on behalf of the authorised subjects, import of user data with security attributes is defined by "*Import of user data with security attributes (FDP_ITC.2) [MIFARE]*".
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 "*Cryptographic key destruction (FCS_CKM.4) [MIFARE]*".
These nine SFRs together provide an access control mechanism as required by the objective *O.Access-Control*.

Security objective “Authentication for DESFire (*O.Authentication*)”

293 The justification related to the security objective “Authentication for DESFire (*O.Authentication*)” is as follows:

294 The two security functional requirements "*Cryptographic operation (FCS_COP.1)*[DES]" and "*Cryptographic operation (FCS_COP.1)*[AES]" require that the TOE provides the basic cryptographic algorithms that can be used to perform the authentication.
The security functional requirements "*User identification before any action (FIA_UID.2) [MIFARE]*", "*User authentication before any action (FIA_UAU.2) [MIFARE]*" and "*Multiple authentication mechanisms (FIA_UAU.5) [MIFARE]*" together define that users must be identified and authenticated before any action. The ‘none’ authentication of "*Multiple authentication mechanisms (FIA_UAU.5) [MIFARE]*" also ensures that a specific subject is identified and authenticated before an explicit authentication request is sent to the TOE.
"*Trusted path (FTP_TRP.1) [MIFARE]*" requires a trusted communication path between the TOE and remote users; FTP_TRP.1.3 especially requires “authentication requests”.
Together with "*Replay detection (FPT_RPL.1) [MIFARE]*" which requires a replay detection for these authentication requests, the seven security functional requirements fulfil the objective *O.Authentication*.

Security objective “DESFire Confidential Communication (*O.Confidentiality*)”

295 The justification related to the security objective “DESFire Confidential communication (*O.Confidentiality*)” is as follows:

296 The two security functional requirements "*Cryptographic operation (FCS_COP.1)*[DES]" and "*Cryptographic operation (FCS_COP.1)*[AES]" require that the TOE provides the basic cryptographic algorithm AES that can be used to protect the communication by encryption.
"*Trusted path (FTP_TRP.1) [MIFARE]*" requires a trusted communication path between the TOE and remote users; FTP_TRP.1.3 especially requires “confidentiality and/or data integrity verification for data transfers protected with AES and based on a setting in the file

attributes".
Together with "*Replay detection (FPT_RPL.1) [MIFARE]*" which requires a replay detection for these data transfers, the three security functional requirements fulfil the objective *O.Confidentiality*.

Security objective "DESFire Data type consistency (*O.Type-Consistency*)"

297 The justification related to the security objective "DESFire Data type consistency (*O.Type-Consistency*)" is as follows:

298 The security functional requirement "*Inter-TSF basic TSF data consistency (FPT_TDC.1) [MIFARE]*" requires the TOE to consistently interpret data files and values. The TOE will honour the respective file formats and boundaries (i.e. upper and lower limits, size limitations). This meets the objective *O.Type-Consistency*.

Security objective "DESFire Transaction mechanism (*O.Transaction*)"

299 The justification related to the security objective "DESFire Transaction mechanism (*O.Transaction*)" is as follows:

300 The security functional requirement "*Basic rollback (FDP_ROL.1) [MIFARE]*" 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*.

Security objective "Preventing traceability for DESFire (*O.No-Trace*)"

301 The justification related to the security objective "Preventing traceability for DESFire (*O.No-Trace*)" is as follows:

302 The security functional requirement "*Unlinkability (FPR_UNL.1) [MIFARE]*" 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*.

Security objective "Usage of hardware platform (*O.Plat-Appl*)"

303 The justification related to the security objective "Usage of hardware platform (*O.Plat-Appl*)" is as follows:

304 The objective was translated from an environment objective in the PP into a TOE objective in this ST. Its goal is to ensure that the hardware platform is used in a secure manner, which is based on the insight that hardware and software have to supplement each other in order to build a secure whole. The ST claims conformance to the PP and the PP SFRs do cover the PP TOE objectives. The PP uses the environment objective OE.Plat-Appl to ensure appropriate software support for its SFRs, but since the TOE does now consist of hardware and software, the PP SFRs do also apply to the Security IC Embedded Software included in the TOE, and thereby all PP SFRs fulfil the objective O.Plat-Appl. In other words: the software support required by the hardware-focused PP is now included in this combined hardware-software TOE and both hardware and software fulfil the PP SFRs.

Security objective "Treatment of user data (*O.Resp-Appl*)"

305 The justification related to the security objective "Treatment of user data (*O.Resp-Appl*)" is as follows:

306 The objective was translated from an environment objective in the PP into a TOE objective in this ST. The objective is that "Security relevant User Data (especially cryptographic keys)

are treated by the Security IC Embedded Software as required by the security needs of the specific application context." The application context is defined by the security environment described in this ST. The additional SFRs defined in this ST do address the additional TOE objectives of the ST based on the ST security environment, therefore *O.Resp-Appl* is fulfilled by the additional ST SFRs.

Security objective "NVM resource availability for DESFire (*O.Resource*)"

307 The justification related to the security objective "Resource availability for DESFire (*O.Resource*)" is as follows:

308 The security functional requirement "*Minimum and maximum quotas (FRU_RSA.2) [MIFARE]*" requires that sufficient parts of the NVM and RAM are reserved for DESFire use. This fulfils the objective *O.Resource*.

Security objective "DESFire code integrity check (*O.Verification*)"

309 The justification related to the security objective "DESFire code integrity check (*O.Verification*)" is as follows:

310 The security functional requirement "*TSF testing (FPT_TST.1) [MIFARE]*" requires that the TSF runs a suite of self tests to demonstrate the correct operation of DESFire. This meets the objective *O.Verification*.

Security objective "DESFire firewall (*O.Firewall*)"

311 The justification related to the security objective "DESFire firewall (*O.Firewall*)" is as follows:

312 The security functional requirements "*Subset access control (FDP_ACC.1) [MIFARE_FWL]*" and "*Security attribute based access control (FDP_ACF.1) [MIFARE_FWL]*", supported by "*Static attribute initialisation (FMT_MSA.3) [MIFARE_FWL]*", require that no application can read, write, compare any piece of data or code belonging to DESFire. This meets the objective *O.Firewall*.

Security objective "DESFire data cleaning for resource sharing (*O.Shr-Res*)"

313 The justification related to the security objective "DESFire data cleaning for resource sharing (*O.Shr-Res*)" is as follows:

314 The security functional requirement "*Subset residual information protection (FDP_RIP.1) [MIFARE]*" requires that the information content of a resource is made unavailable upon its deallocation from DESFire. This meets the objective *O.Shr-Res*.

7.4.3 Additional security requirements are consistent

"Cryptographic operation (***FCS_COP.1***) & key generation (***FCS_CKM.1***)"

315 These security requirements have already been argued in *Section : Security objective "Additional Specific Security Functionality (AUG1.O.Add-Functions)"* above.

**"Static attribute initialisation (*FMT_MSA.3 [Memories]*),
Management of security attributes (*FMT_MSA.1 [Memories]*),
Complete access control (*FDP_ACC.2 [Memories]*),
Security attribute based access control (*FDP_ACF.1 [Memories]*)"**

316 These security requirements have already been argued in *Section : Security objective "Dynamic Area based Memory Access Control (AUG4.O.Mem-Access)"* above.

**"Security roles (*FMT_SMR.1 [MIFARE]*),
Subset access control (*FDP_ACC.1 [MIFARE]*),
Security attribute based access control (*FDP_ACF.1 [MIFARE]*),
Static attribute initialisation (*FMT_MSA.3 [MIFARE]*),
Management of security attributes (*FMT_MSA.1 [MIFARE]*),
Specification of TSF data (*FMT_MTD.1 [MIFARE]*)
Specification of management function (*FMT_SMF.1 [MIFARE]*)
Import of user data with security attributes (*FDP_ITC.2 [MIFARE]*)
Cryptographic key destruction (*FCS_CKM.4 [MIFARE]*)"**

317 These security requirements have already been argued in *Section : Security objective "Access control for DESFire (O.Access-Control)"* above.

**"User identification before any action (*FIA_UID.2 [MIFARE]*),
User authentication before any action (*FIA_UAU.2 [MIFARE]*),
Multiple authentication mechanisms (*FIA_UAU.5 [MIFARE]*)"**

318 These security requirements have already been argued in *Section : Security objective "Authentication for DESFire (O.Authentication)"* above.

**"Trusted path (*FPT_TRP.1 [MIFARE]*),
Replay detection (*FPT_RPL.1 [MIFARE]*)"**

319 These security requirements have already been argued in *Section : Security objective "DESFire Confidential Communication (O.Confidentiality)"* above.

"Inter-TSF basic TSF data consistency (*FPT_TDC.1 [MIFARE]*)"

320 This security requirement has already been argued in *Section : Security objective "DESFire Data type consistency (O.Type-Consistency)"* above.

"Basic rollback (*FDP_ROL.1 [MIFARE]*)"

321 This security requirement has already been argued in *Section : Security objective "DESFire Transaction mechanism (O.Transaction)"* above.

"Unlinkability (*FPR_UNL.1 [MIFARE]*)"

322 This security requirement has already been argued in *Section : Security objective "Preventing traceability for DESFire (O.No-Trace)"* above.

"Minimum and maximum quotas (*FRU_RSA.2 [MIFARE]*)"

323 This security requirement has already been argued in *Section : Security objective "NVM resource availability for DESFire (O.Resource)"* above.

"TSF testing (*FPT_TST.1 [MIFARE]*)"

324 This security requirement has already been argued in *Section : Security objective "DESFire code integrity check (O.Verification)"* above.

"Subset access control (*FDP_ACC.1 [MIFARE_FWL]*),
Security attribute based access control (*FDP_ACF.1 [MIFARE_FWL]*),
Static attribute initialisation (*FMT_MSA.3 [MIFARE_FWL]*),

325 These security requirements have already been argued in *Section : Security objective "DESFire firewall (O.Firewall)"* above.

"Subset residual information protection (*FDP_RIP.1 [MIFARE]*)"

326 This security requirement has already been argued in *Section : Security objective "DESFire data cleaning for resource sharing (O.Shr-Res)"* above.

7.4.4 Dependencies of Security Functional Requirements

327 All dependencies of Security Functional Requirements have been fulfilled in this Security Target except :

- those justified in the *BSI-PP-0035* protection profile security requirements rationale,
- those justifed in *AUG* security requirements rationale (except on FMT_MSA.2, see discussion below),
- the dependency of *FCS_COP.1* and *FCS_CKM.1* on FCS_CKM.4 (see discussion below).
- the dependency of *FMT_MSA.3 [MIFARE_FWL]* on FMT_MSA.1 and FMT_SMR.1 (see discussion below).

328 Details are provided in *Table 12* below.

Table 12. Dependencies of security functional requirements

Label	Dependencies	Fulfilled by security requirements in this Security Target	Dependency already in *BSI-PP-0035* or in *AUG*
FRU_FLT.2	FPT_FLS.1	Yes	Yes, *BSI-PP-0035*
FPT_FLS.1	None	No dependency	Yes, *BSI-PP-0035*
FMT_LIM.1	FMT_LIM.2	Yes	Yes, *BSI-PP-0035*
FMT_LIM.2	FMT_LIM.1	Yes	Yes, *BSI-PP-0035*
FAU_SAS.1	None	No dependency	Yes, *BSI-PP-0035*
FPT_PHP.3	None	No dependency	Yes, *BSI-PP-0035*
FDP_ITT.1	FDP_ACC.1 or FDP_IFC.1	Yes	Yes, *BSI-PP-0035*
FPT_ITT.1	None	No dependency	Yes, *BSI-PP-0035*
FDP_IFC.1	FDP_IFF.1	No, see *BSI-PP-0035*	Yes, *BSI-PP-0035*
FCS_RNG.1	None	No dependency	Yes, *BSI-PP-0035*

Table 12. Dependencies of security functional requirements (continued)

Label	Dependencies	Fulfilled by security requirements in this Security Target	Dependency already in *BSI-PP-0035* or in *AUG*
FCS_COP.1	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	Yes, by FDP_ITC.1 and FCS_CKM.1, see discussion below	Yes, *AUG* ***#1***
	FCS_CKM.4	No, see discussion below	
FCS_CKM.1	[FDP_CKM.2 or FCS_COP.1]	Yes, by FCS_COP.1	
	FCS_CKM.4	No, see discussion below	
FDP_ACC.2 [Memories]	FDP_ACF.1 [Memories]	Yes	***No,*** *CCMB-2012-09-002*
FDP_ACF.1 [Memories]	FDP_ACC.1 [Memories]	Yes, by FDP_ACC.2 [Memories]	Yes, *AUG* ***#4***
	FMT_MSA.3 [Memories]	Yes	
FMT_MSA.3 [Memories]	FMT_MSA.1 [Memories]	Yes	Yes, *AUG* ***#4***
	FMT_SMR.1 [Memories]	No, see *AUG* ***#4***	
FMT_MSA.1 [Memories]	[FDP_ACC.1 [Memories] or FDP_IFC.1]	Yes, by FDP_ACC.2 [Memories] and FDP_IFC.1	Yes, *AUG* ***#4***
	FMT_SMF.1 [Memories]	Yes	***No,*** *CCMB-2012-09-002*
	FMT_SMR.1 [Memories]	No, see *AUG* ***#4***	Yes, *AUG* ***#4***
FMT_SMF.1 [Memories]	None	No dependency	***No,*** *CCMB-2012-09-002*
FMT_SMR.1 [MIFARE]	FIA_UID.1 [MIFARE]	Yes, by FIA_UID.2 [MIFARE]	***No,*** *CCMB-2012-09-002*
FDP_ACC.1 [MIFARE]	FDP_ACF.1 [MIFARE]	Yes	***No,*** *CCMB-2012-09-002*
FDP_ACF.1 [MIFARE]	FDP_ACC.1 [MIFARE]	Yes	***No,*** *CCMB-2012-09-002*
	FMT_MSA.3 [MIFARE]	Yes	
FMT_MSA.3 [MIFARE]	FMT_MSA.1 [MIFARE]	Yes	***No,*** *CCMB-2012-09-002*
	FMT_SMR.1 [MIFARE]	Yes	

Table 12. Dependencies of security functional requirements (continued)

Label	Dependencies	Fulfilled by security requirements in this Security Target	Dependency already in *BSI-PP-0035* or in *AUG*
FMT_MSA.1 [MIFARE]	[FDP_ACC.1 [MIFARE] or FDP_IFC.1]	Yes, by FDP_ACC.1 [MIFARE]	***No,*** *CCMB-2012-09-002*
	FMT_SMF.1 [MIFARE]	Yes	
	FMT_SMR.1 [MIFARE]	Yes	
FMT_SMF.1 [MIFARE]	None	No dependency	***No,*** *CCMB-2012-09-002*
FDP_ITC.2 [MIFARE]	FDP_ACC.1 [MIFARE] or FDP_IFC.1	Yes, by FDP_ACC.1 [MIFARE]	***No,*** *CCMB-2012-09-002*
	FTP_ITC.1 or FPT_TRP.1 [MIFARE]	Yes, by FPT_TRP.1 [MIFARE]	
	FPT_TDC.1 [MIFARE]	Yes	
FPT_TDC.1 [MIFARE]	None	No dependency	***No,*** *CCMB-2012-09-002*
FIA_UID.2 [MIFARE]	None	No dependency	***No,*** *CCMB-2012-09-002*
FIA_UAU.2 [MIFARE]	FIA_UID.1	Yes, by FIA_UID.2 [MIFARE]	***No,*** *CCMB-2012-09-002*
FIA_UAU.5 [MIFARE]	None	No dependency	***No,*** *CCMB-2012-09-002*
FMT_MTD.1 [MIFARE]	FMT_SMR.1 [MIFARE]	Yes	***No,*** *CCMB-2012-09-002*
	FMT_SMF.1 [MIFARE]	Yes	
FPT_TRP.1 [MIFARE]	None	No dependency	***No,*** *CCMB-2012-09-002*
FCS_CKM.4 [MIFARE]	[FDP_ITC.1 or FDP_ITC.2 [MIFARE] or FCS_CKM.1]	Yes, by FDP_ITC.2 [MIFARE]	***No,*** *CCMB-2012-09-002*
FDP_ROL.1 [MIFARE]	FDP_ACC.1 [MIFARE] or FDP_IFC.1	Yes, by FDP_ACC.1 [MIFARE]	***No,*** *CCMB-2012-09-002*
FPT_RPL.1 [MIFARE]	None	No dependency	***No,*** *CCMB-2012-09-002*
FPR_UNL.1 [MIFARE]	None	No dependency	***No,*** *CCMB-2012-09-002*

Table 12. Dependencies of security functional requirements (continued)

Label	Dependencies	Fulfilled by security requirements in this Security Target	Dependency already in *BSI-PP-0035* or in *AUG*
FPT_TST.1 [MIFARE]	None	No dependency	***No,*** *CCMB-2012-09-002*
FRU_RSA.2 [MIFARE]	None	No dependency	***No,*** *CCMB-2012-09-002*
FDP_ACC.1 [MIFARE_FWL]	FDP_ACF.1 [MIFARE_FWL]	Yes	***No,*** *CCMB-2012-09-002*
FDP_ACF.1 [MIFARE_FWL]	FDP_ACC.1 [MIFARE_FWL]	Yes	***No,*** *CCMB-2012-09-002*
	FMT_MSA.3 [MIFARE_FWL]	Yes	
FMT_MSA.3 [MIFARE_FWL]	FMT_MSA.1	No, see discussion below	***No,*** *CCMB-2012-09-002*
	FMT_SMR.1	No, see discussion below	
FDP_RIP.1 [MIFARE]	None	No dependency	***No,*** *CCMB-2012-09-002*

329 Part 2 of the Common Criteria defines the dependency of "Cryptographic operation (FCS_COP.1)" on "Import of user data without security attributes (FDP_ITC.1)" or "Import of user data with security attributes (FDP_ITC.2)" or "Cryptographic key generation (FCS_CKM.1)". In this particular TOE, "Cryptographic key generation (FCS_CKM.1)" may be used for the purpose of creating cryptographic keys, but also, the ES has all possibilities to implement its own creation function, in conformance with its security policy.

330 Part 2 of the Common Criteria defines the dependency of "Cryptographic operation (FCS_COP.1)" and "Cryptographic key generation (FCS_CKM.1)" on "Cryptographic key destruction (FCS_CKM.4)". In this particular TOE, there is no specific function for the destruction of the keys. The ES has all possibilities to implement its own destruction function, in conformance with its security policy. Therefore, FCS_CKM.4 is not defined in this ST.

331 Part 2 of the Common Criteria defines the dependency of "Static attribute initialisation (FMT_MSA.3) [MIFARE]" on "Management of security attributes (FMT_MSA.1)" and "Security roles (FMT_SMR.1)". For this particular instantiation of the access control attributes aimed at protecting DESFire code and data from unauthorised accesses, the security attributes are only static, initialized at product start. Therefore, there is no need to identify management capabilities and associated roles in form of Security Functional Requirements "FMT_MSA.1" and "FMT_SMR.1".

7.4.5 Rationale for the Assurance Requirements

Security assurance requirements added to reach EAL5 (*Table 10*)

332 Regarding application note 21 of *BSI-PP-0035*, this Security Target chooses EAL5 because developers and users require a high level of independently assured security in a planned development and require a rigorous development approach without incurring unreasonable costs attributable to specialist security engineering techniques.

333 EAL5 represents a meaningful increase in assurance from EAL4 by requiring semiformal design descriptions, a more structured (and hence analyzable) architecture, and improved mechanisms and/or procedures that provide confidence that the TOE will not be tampered during development.

334 The assurance components in an evaluation assurance level (EAL) 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 and the internal consistency is still guaranteed.

335 Note that detailed and updated refinements for assurance requirements are given in *Section 7.3*.

Dependencies of assurance requirements

336 Dependencies of security assurance requirements are fulfilled by the EAL5 package selection.

337 Augmentation to this package are identified in paragraph *263* and do not introduce dependencies not already satisfied by the EAL5 package.

8 TOE summary specification

338 This section demonstrates how the TOE meets each Security Functional Requirement, which will be further detailed in the ADV_FSP documents.

339 The complete TOE summary specification has been presentad and evaluated in the ST31 - K330A version F (dual or contactless mode only) with optional cryptographic library Neslib 3.2, and optional technology MIFARE DESFire™ EV1 2.2 SECURITY TARGET.

340 For confidentiality reasons, the TOE summary specification is not fully reproduced here.

8.1 Limited fault tolerance (FRU_FLT.2)

341 The TSF provides limited fault tolerance, by managing a certain number of faults or errors that may happen, related to memory contents, CPU, random number generation and cryptographic operations, thus preventing risk of malfunction.

8.2 Failure with preservation of secure state (FPT_FLS.1)

342 The TSF provides preservation of secure state by detecting and managing the following events, resulting in an immediate reset:

- Die integrity violation detection,
- Errors on memories,
- Glitches,
- High voltage supply,
- CPU errors,
- MPU errors,
- External clock incorrect frequency,
- etc..

343 The ES can generate a software reset.

8.3 Limited capabilities (FMT_LIM.1)

344 The TSF ensures that only very limited test capabilities are available in USER configuration, in accordance with SFP_1: Limited capability and availability Policy.

8.4 Limited availability (FMT_LIM.2)

345 The TOE is either in TEST,or USER configuration.

346 The only authorised TOE configuration modification is:

- TEST to USER configuration.

347 The TSF ensures the switching and the control of TOE configuration.

348 The TSF reduces the available features depending on the TOE configuration.

8.5 Audit storage (FAU_SAS.1)

349 In User configuration, the TOE provides commands to store data and/or pre-personalisation data and/or supplements of the ES in the NVM. These commands are only available to authorized processes, and only until phase 6.

8.6 Resistance to physical attack (FPT_PHP.3)

350 The TSF ensures resistance to physical tampering, thanks to the following features:

- The TOE implements counter-measures that reduce the exploitability of physical probing.
- The TOE is physically protected by an active shield that commands an automatic reaction on die integrity violation detection.

8.7 Basic internal transfer protection (FDP_ITT.1), Basic internal TSF data transfer protection (FPT_ITT.1) & Subset information flow control (FDP_IFC.1)

351 The TSF prevents the disclosure of internal and user data thanks to:

- Memories scrambling and encryption,
- Bus encryption,
- Mechanisms for operation execution concealment,
- etc..

8.8 Random number generation (FCS_RNG.1)

352 The TSF provides 8-bit true random numbers that can be qualified with the test metrics required by the BSI-AIS31 standard for a P2 class device.

8.9 Cryptographic operation: DES / 3DES operation (FCS_COP.1 [EDES])

353 The TOE provides an EDES accelerator that has the capability to perform DES and Triple DES encryption and decryption conformant to FIPS PUB 46-3.

354 The EDES accelerator offers a Cipher Block Chaining (CBC) mode conformant to ISO/IEC 10116, and a Cipher Block Chaining Message Authentication Code (CBC-MAC) mode conformant to ISO/IEC 9797-1.

8.10 Cryptographic operation: AES operation (FCS_COP.1 [AES])

355 The AES accelerator provides the following standard AES cryptographic operations for key sizes of 128, 192 and 256 bits, conformant to FIPS PUB 197 with intrinsic counter-measures against attacks:

- randomize,
- key expansion,
- cipher,
- inverse cipher.

356 If Neslib is embedded, the cryptographic library Neslib provides the same standard AES cryptographic operations.

8.11 Cryptographic operation: RSA operation (FCS_COP.1 [RSA]) if Neslib only

357 The cryptographic library Neslib provides the RSA public key cryptographic operation for modulus sizes up to 4096 bits, conformant to PKCS #1 V2.1.

358 The cryptographic library Neslib provides the RSA private key cryptographic operation with or without CRT for modulus sizes up to 4096 bits, conformant to PKCS #1 V2.1.

8.12 Cryptographic operation: Elliptic Curves Cryptography operation (FCS_COP.1 [ECC]) if Neslib only

359 The cryptographic library Neslib provides to the ES developer the following efficient basic functions for Elliptic Curves Cryptography over prime fields, all conformant to IEEE 1363-2000 and IEEE 1363a-2004, including:

- private scalar multiplication,
- preparation of Elliptic Curve computations in affine coordinates,
- public scalar multiplication,
- point validity check.

8.13 Cryptographic operation: SHA operation (FCS_COP.1 [SHA]) if Neslib only

360 The cryptographic library Neslib provides the SHA-1, SHA-224, SHA-256, SHA-384, SHA-512 secure hash functions conformant to FIPS PUB 180-1, FIPS PUB 180-2, ISO/IEC 10118-3:1998.

361 The cryptographic library Neslib provides the SHA-1 secure hash function conformant to FIPS PUB 180-1, FIPS PUB 180-2, ISO/IEC 10118-3:1998, and offering resistance against side channel and fault attacks.

8.14 Cryptographic key generation: Prime generation (FCS_CKM.1 [Prime_generation]) & Cryptographic key generation: Protected prime generation (FCS_CKM.1 [Protected_prime_generation]) if Neslib only

362 The cryptographic library Neslib provides prime numbers generation for key sizes up to 2048 bits conformant to FIPS PUB 140-2 and FIPS PUB 186, and offering resistance against side channel and fault attacks.

8.15 Cryptographic key generation: RSA key generation (FCS_CKM.1 [RSA_key_generation]) & Cryptographic key generation: Protected RSA key generation (FCS_CKM.1 [Protected_RSA_key_generation]) if Neslib only

363 The cryptographic library Neslib provides standard RSA public and private key computation for key sizes upto 4096 bits conformant to FIPS PUB 140-2, ISO/IEC 9796-2 and PKCS #1 V2.1, and offering resistance against side channel and fault attacks.

8.16 Static attribute initialisation (FMT_MSA.3) [Memories]

364 The TOE enforces a default memory protection policy when none other is programmed by the ES.

8.17 Management of security attributes (FMT_MSA.1) [Memories] & Specification of management functions (FMT_SMF.1) [Memories]

365 The TOE provides a dynamic Memory Protection Unit (MPU), that can be configured by the ES.

8.18 Complete access control (FDP_ACC.2) [Memories] & Security attribute based access control (FDP_ACF.1) [Memories]

366 The TOE enforces the dynamic memory protection policy for data access and code access thanks to a dynamic Memory Protection Unit (MPU), programmed by the ES. Overriding the MPU set of access rights, the TOE enforces additional protections on specific parts of the memories.

8.19 Security roles (FMT_SMR.1) [MIFARE]

367 DESFire supports the assignment of roles to users through the assignment of different keys for the different roles and through the structure and configuration of the access rights. This

allows to distinguish between the roles of Administrator, Application Manager, Application User, and Everybody.

8.20 Subset access control (FDP_ACC.1) [MIFARE]

368 For each DESFire command subject to access control, the DESFire library verifies if the DESFire access conditions are satisfied and returns an error when this is not the case.

8.21 Security attribute based access control (FDP_ACF.1) [MIFARE]

369 The DESFire library verifies the DESFire security attributes during the execution of DESFire commands to enforce the Access Control Policy defined by the DESFire interface specification.

8.22 Static attribute initialisation (FMT_MSA.3) [MIFARE]

370 The DESFire library initialises all the static attributes to the values defined by DESFire interface specifications before they can be used by the Embedded Software.

8.23 Management of security attributes (FMT_MSA.1) [MIFARE]

371 The DESFire library verifies the DESFire security attributes during the execution of DESFire commands to enforce the Access Control Policy on the security attributes.

8.24 Specification of Management Functions (FMT_SMF.1) [MIFARE]

372 The DESFire library implements the management functions defined by the DESFire interface specifications for authentication, changing security attributes and creating or deleting an application, a value or a data file.

8.25 Import of user data with security attributes (FDP_ITC.2) [MIFARE]

373 The DESFire library implements the DESFire interface specifications and enforces the Access Control Policy to associate the user data to the security attributes.

8.26 Inter-TSF basic TSF data consistency (FPT_TDC.1) [MIFARE]

374 The DESFire library implements the DESFire interface specifications, supporting consistent interpretation and modification control of inter-TSF exchanges.

8.27 Cryptographic key destruction (FCS_CKM.4) [MIFARE]

375 The DESFire library erases key values from memory after their context becomes obsolete.

8.28 User identification before any action (FIA_UID.2) [MIFARE]

376 The DESFire library identifies the user through the key selected for authentication as specified by the DESFire Interface Specification.

8.29 User authentication before any action (FIA_UAU.2) [MIFARE]

377 During the authentication, the DESFire library verifies that the user knows the selected key.

378 After this authentication, both parties share a session key.

8.30 Multiple authentication mechanisms (FIA_UAU.5) [MIFARE]

379 The DESFire library implements the DESFire Interface Specification, that has a mechanism to authenticate Administrator, Application Manager and Application User, while Everybody is assumed when there is no valid authentication state.

380 Two types of authentication are supported: the native DESFire 3-pass authentication and the ISO authentication.

8.31 Management of TSF data (FMT_MTD.1) [MIFARE]

381 The DESFire library implements the DESFire Interface Specification, restricting key modifications in ways configurable through the security attributes to authenticated users, or disabling key modification capabilities.

8.32 Trusted path (FTP_TRP.1) [MIFARE]

382 The DESFire library implements the DESFire Interface Specification allowing to establish and enforce a trusted path between itself and remote users.

8.33 Basic rollback (FDP_ROL.1) [MIFARE]

383 The DESFire library implements the DESFire transaction mechanism ensuring that either all or none of the (modifying) file commands within a transaction are performed. If not, they are rolled back.

8.34 Replay detection (FPT_RPL.1) [MIFARE]

384 The DESFire library implements the DESFire authentication command, and authenticated commands, that allow replay detection.

8.35 Unlinkability (FPR_UNL.1) [MIFARE]

385 DESFire provides an Administrator option to use random UID during the ISO 14443 anti-collision sequence, preventing the traceability through UID. At higher level, the DESFire access control - when configured for this purpose - provides traceability protection.

8.36 TSF testing (FPT_TST.1) [MIFARE]

386 The DESFire library performs a code integrity test before starting execution of DESFire commands. This integrity check can also be performed on request of the Embedded Software.

8.37 Minimum and maximum quotas (FRU_RSA.2) [MIFARE]

387 The DESFire library ensures the memory required for its operation is available.

8.38 Subset residual information protection (FDP_RIP.1) [MIFARE]

388 At the end of commands execution or upon interrupt, the DESFire library cleans the confidential data from crypto-processors and CPU registers it uses.

8.39 Subset access control (FDP_ACC.1) [MIFARE_FWL] & Security attribute based access control (FDP_ACF.1) [MIFARE_FWL]

389 The Library Protection Unit is used to isolate the DESFire firmware (code and data) from the rest of the code embedded in the device.

8.40 Static attribute initialisation (FMT_MSA.3) [MIFARE_FWL]

390 At product start, all the static attributes are initialised, which are needed to protect the segments where DESFire code and data are stored.

9 References

391 Protection Profile references

Component description	Reference	Revision
Security IC Platform Protection Profile	BSI-PP-0035	1.0

392 ST31 - K330A Security Target reference

Component description	Reference
ST31 - K330A version F (dual or contactless mode only) with optional cryptographic library Neslib 3.2, and optional technology MIFARE DESFire™ EV1 2.2 SECURITY TARGET	SMD_MR31Zxxx_ST_13_001

393 Guidance documentation references

Component description	Reference	Revision
ST31 - K330 platform - Sx31Zxxx, Mx31Zxxx - Secure dual interface microcontrolleer with enhanced security and up to 52 Kbytes of EEPROM - Datasheet	DS_SR31Z052	1.0
ST31 - K330 platform 90nm F10 CMOS die description	DD_31Z052	2
ARM SC000 Technical Reference Manual - R0P0	ARM DDI 0456	A
ARM v6-M Architecture Reference Manual	ARM DDI 0419	C
ST31 - K330 Security guidance	AN_SECU_ST31_K330	2.0
ST31 - AIS31 Compliant Random Number user manual	UM_31_AIS31	2
ST31 - AIS31 Reference implementation - Startup, online and total failure tests - Application Note	AN_31_AIS31	2
ST31 Secure MCUs NesLib 3.2 cryptographic library - User manual	UM_31_NESLIB_3.2	6
Application Note - Recommendations for use of EEPROM and code signature in ST31-K330 secure microcontrollers	AN_31_EEPROM	1
ST31-K330 and ST33-K8H0 secure microcontrollers - Power supply glitch detector characteristics	AN_31_GLITCH	2
Application note - ST31-K330 Dual Interface Secure MCUs - Recommandation for contactless operations	AN_31_RCMD	1
User manual: MIFARE DESFire EV1 library 2.2	UM_MIFARE_DESFire_EV1_2.2	2
MIFARE DESFire EV1 interface specification - User Manual	UM_MIFARE_DESFire_EV1_Interface	2

394 Standards references

Ref	Identifier	Description
[1]	BSI-AIS31	A proposal for functionality classes and evaluation methodology for true (physical) random number generators, W. Killmann & W. Schindler BSI, Version 3.1, 25-09-2001
[2]	FIPS PUB 46-3	FIPS PUB 46-3, Data encryption standard (DES), National Institute of Standards and Technology, U.S. Department of Commerce, 1999
[3]	FIPS PUB 140-2	FIPS PUB 140-2, Security Requirements for Cryptographic Modules, National Institute of Standards and Technology, U.S. Department of Commerce, 1999
[4]	FIPS PUB 180-1	FIPS PUB 180-1 Secure Hash Standard, National Institute of Standards and Technology, U.S. Department of Commerce,1995
[5]	FIPS PUB 180-2	FIPS PUB 180-2 Secure Hash Standard with Change Notice 1 dated February 25,2004, National Institute of Standards and Technology, U.S.A., 2004
[6]	FIPS PUB 186	FIPS PUB 186 Digital Signature Standard (DSS), National Institute of Standards and Technology, U.S.A., 1994
[7]	FIPS PUB 197	FIPS PUB 197, Advanced Encryption Standard (AES), National Institute of Standards and Technology, U.S. Department of Commerce, November 2001
[8]	ISO/IEC 9796-2	ISO/IEC 9796, Information technology - Security techniques - Digital signature scheme giving message recovery - Part 2: Integer factorization based mechanisms, ISO, 2002
[9]	ISO/IEC 9797-1	ISO/IEC 9797, Information technology - Security techniques - Message Authentication Codes (MACs) - Part 1: Mechanisms using a block cipher, ISO, 1999
[10]	ISO/IEC 10116	ISO/IEC 10116, Information technology - Security techniques - Modes of operation of an n-bit block cipher algorithm, ISO, 1997
[11]	ISO/IEC 10118-3:1998	ISO/IEC 10118-3:1998, Information technology - Security techniques - Hash functions - Part 3: Dedicated hash functions
[12]	ISO/IEC 14888	ISO/IEC 14888, Information technology - Security techniques - Digital signatures with appendix - Part 1: General (1998), Part 2: Identity-based mechanisms (1999), Part 3: Certificate based mechanisms (2006), ISO
[13]	CCMB-2012-09-001	Common Criteria for Information Technology Security Evaluation - Part 1: Introduction and general model, September 2012, version 3.1 Revision 4
[14]	CCMB-2012-09-002	Common Criteria for Information Technology Security Evaluation - Part 2: Security functional components, September 2012, version 3.1 Revision 4
[15]	CCMB-2012-09-003	Common Criteria for Information Technology Security Evaluation - Part 3: Security assurance components, September 2012, version 3.1 Revision 4
[16]	AUG	Smartcard Integrated Circuit Platform Augmentations, Atmel, Hitachi Europe, Infineon Technologies, Philips Semiconductors, Version 1.0, March 2002.

Ref	Identifier	Description
[17]	MIT/LCS/TR-212	On digital signatures and public key cryptosystems, Rivest, Shamir & Adleman Technical report MIT/LCS/TR-212, MIT Laboratory for computer sciences, January 1979
[18]	IEEE 1363-2000	IEEE 1363-2000, Standard Specifications for Public Key Cryptography, IEEE, 2000
[19]	IEEE 1363a-2004	IEEE 1363a-2004, Standard Specifications for Public Key Cryptography - Amendment 1:Additional techniques, IEEE, 2004
[20]	PKCS #1 V2.1	PKCS #1 V2.1 RSA Cryptography Standard, RSA Laboratories, June 2002
[21]	MOV 97	Alfred J. Menezes, Paul C. van Oorschot and Scott A. Vanstone, Handbook of Applied Cryptography, CRC Press, 1997

Appendix A Glossary

A.1 Terms

Authorised user

A user who may, in accordance with the TSP, perform an operation.

Composite product

Security IC product which includes the Security Integrated Circuit (i.e. the TOE) and the Embedded Software and is evaluated as composite target of evaluation.

End-consumer

User of the Composite Product in Phase 7.

Integrated Circuit (IC)

Electronic component(s) designed to perform processing and/or memory functions.

IC Dedicated Software

IC proprietary software embedded in a Security IC (also known as IC firmware) and developed by ***ST***. Such software is required for testing purpose (IC Dedicated Test Software) but may provide additional services to facilitate usage of the hardware and/or to provide additional services (IC Dedicated Support Software).

IC Dedicated Test Software

That part of the IC Dedicated Software which is used to test the TOE before TOE Delivery but which does not provide any functionality thereafter.

IC developer

Institution (or its agent) responsible for the IC development.

IC manufacturer

Institution (or its agent) responsible for the IC manufacturing, testing, and pre-personalization.

IC packaging manufacturer

Institution (or its agent) responsible for the IC packaging and testing.

Initialisation data

Initialisation Data defined by the TOE Manufacturer to identify the TOE and to keep track of the Security IC's production and further life-cycle phases are considered as belonging to the TSF data. These data are for instance used for traceability and for TOE identification (identification data)

Object

An entity within the TSC that contains or receives information and upon which subjects perform operations.

Packaged IC

Security IC embedded in a physical package such as micromodules, DIPs, SOICs or TQFPs.

Pre-personalization data

Any data supplied by the Card Manufacturer that is injected into the non-volatile memory by the Integrated Circuits manufacturer (Phase 3). These data are for instance used for traceability and/or to secure shipment between phases.

Secret

Information that must be known only to authorised users and/or the TSF in order to enforce a specific SFP.

Security IC

Composition of the TOE, the Security IC Embedded Software, User Data, and the package.

Security IC Embedded SoftWare (ES)

Software embedded in the Security IC and not developed by the IC designer. The Security IC Embedded Software is designed in Phase 1 and embedded into the Security IC in Phase 3.

Security IC embedded software (ES) developer

Institution (or its agent) responsible for the security IC embedded software development and the specification of IC pre-personalization requirements, if any.

Security attribute

Information associated with subjects, users and/or objects that is used for the enforcement of the TSP.

Sensitive information

Any information identified as a security relevant element of the TOE such as:

– the application data of the TOE (such as IC pre-personalization requirements, IC and system specific data),
– the security IC embedded software,
– the IC dedicated software,
– the IC specification, design, development tools and technology.

Smartcard

A card according to ISO 7816 requirements which has a non volatile memory and a processing unit embedded within it.

Subject

An entity within the TSC that causes operations to be performed.

Test features

All features and functions (implemented by the IC Dedicated Software and/or hardware) which are designed to be used before TOE Delivery only and delivered as part of the TOE.

TOE Delivery

The period when the TOE is delivered which is after Phase 3 ***or Phase 4 in this Security target***.

TSF data

Data created by and for the TOE, that might affect the operation of the TOE.

User

Any entity (human user or external IT entity) outside the TOE that interacts with the TOE.

User data

All data managed by the Smartcard Embedded Software in the application context. User data comprise all data in the final Smartcard IC except the TSF data.

A.2 Abbreviations

Table 13. List of abbreviations

Term	Meaning
AIS	Application notes and Interpretation of the Scheme (BSI)
ALU	Arithmetical and Logical Unit.
BSI	Bundesamt für Sicherheit in der Informationstechnik.
CBC	Cipher Block Chaining.
CBC-MAC	Cipher Block Chaining Message Authentication Code.
CC	Common Criteria Version 3.1.
CPU	Central Processing Unit.
CRC	Cyclic Redundancy Check.
DCSSI	Direction Centrale de la Sécurité des Systèmes d'Information
DES	Data Encryption Standard.
DIP	Dual-In-Line Package.
EAL	Evaluation Assurance Level.
ECB	Electronic Code Book.
EDES	Enhanced DES.
EEPROM	Electrically Erasable Programmable Read Only Memory.
ES	Security IC Embedded SoftWare.
FIPS	Federal Information Processing Standard.
I/O	Input / Output.
IC	Integrated Circuit.
ISO	International Standards Organisation.
IT	Information Technology.
MPU	Memory Protection Unit.
NESCRYPT	Next Step Cryptography Accelerator.
NIST	National Institute of Standards and Technology.
NVM	Non Volatile Memory.
OSP	Organisational Security Policy.
OST	Operating System for Test.
PP	Protection Profile.
PUB	Publication Series.
RAM	Random Access Memory.
RF	Radio Frequency.
RF UART	Radio Frequency Universal Asynchronous Receiver Transmitter.
ROM	Read Only Memory.

Table 13. List of abbreviations (continued)

Term	Meaning
RSA	Rivest, Shamir & Adleman.
SAR	Security Assurance Requirement.
SFP	Security Function Policy.
SFR	Security Functional Requirement.
SOIC	Small Outline IC.
ST	Context dependent : STMicroelectronics or Security Target.
TOE	Target of Evaluation.
TQFP	Thin Quad Flat Package.
TRNG	True Random Number Generator.
TSC	TSF Scope of Control.
TSF	TOE Security Functionality.
TSFI	TSF Interface.
TSP	TOE Security Policy.
TSS	TOE Summary Specification.

10 Revision history

Table 14. Document revision history

Date	Revision	Changes
19-Mar-2013	01.00	Initial release.
06-Jun-2013	01.01	Change in references.
05-Sep-2013	01.02	Change in references.