Proprietary

M/Chip 4 for MULTOS Application Security Target Summary

Document Number	mxi-mchip-stg-005
Version	1-0
Released date	11 Jul 2003

Archived by: _____ Date: _____

Copyright

History

Version	Date	Change Notes	Author
0-1	27 Jun 2003	First release, based on mxi-mchip-stg-003 v4-1	J Tierney
1-0	11 Jul 2003	Minor updates following formal review.	J Tierney

Preface

Objectives

The objective of this document is to summarise the security target for the Common Criteria Evaluation of the M/Chip 4 for MULTOS Application.

Scope

- This Security Target summary is limited to the M/Chip 4 for MULTOS application (version 1.0.1.1) only.

- This Security Target summary does not cover chip embedding, MULTOS enablement or M/Chip 4 for MULTOS application loading (which includes customisation and personalisation) prior to delivery to the end-user. Key load is included in the application load process and hence is outside the scope of the evaluation, as is the platform upon which the application will run.

- The M/Chip 4 for MULTOS application can support two application profiles (M/Chip Select 4 and M/Chip Lite 4), which are purely data dependent. The evaluation, and subsequent certification, of the M/Chip 4 for MULTOS application will therefore ensure that both application profiles that can be data configured, but use version 1.0.1.1 of M/Chip 4 for MULTOS are certified to EAL4+.

Audience

Those wishing to obtain a summary of the Security Target used for the evaluation of the M/Chip 4 for MULTOS application.

Related documents

Document Title	Doc Number	Ver	Ref
M/Chip 4 for MULTOS Application Assurance Requirements Mapping	mxi-mchip4-doc-001	TBI	ARM
Common Criteria for Information Technology Security Evaluation Part 1: Introduction and General Model	CCIMB-99-031 (also defined as ISO/IEC 15408-1:1999(E))	Version 2.1, August 1999	CC1
Common Criteria for Information Technology Security Evaluation Part 2: Security Functional Requirements	CCIMB-99-032 (also defined as ISO/IEC 15408-2:1999(E))	Version 2.1, August 1999	CC2
Common Criteria for Information Technology Security Evaluation Part 3: Security Assurance Requirements	CCIMB-99-033 (also defined as ISO/IEC 15408-3:1999(E))	Version 2.1, August 1999	CC3
[CC1] + [CC2] + [CC3]	-	-	CC
EMV2000 Integrated Circuit Card Specification for Payment Systems, Book 1 Application Independent ICC to Terminal Interface Requirements	-	4.0, December 2000	EMV1
EMV2000 Integrated Circuit Card Specification for Payment Systems, Book 2 Security and Key Management	-	4.0, December 2000	EMV2
EMV2000 Integrated Circuit Card Specification for Payment Systems, Book 3 Application Specification	-	4.0, December 2000	EMV3
EMV2000 Integrated Circuit Card Specification for Payment Systems, Book 4 Cardholder, Attendant and Acquirer Interface Requirements	-	4.0, December 2000	EMV4
Modification to Combined Dynamic Data Authentication and Application Cryptogram Generation	EMVCo Specification Update: Bulletin No 6	December 14th 2001	EMVB6
Modification to Combined Dynamic Data Authentication	EMVCo Specification Update: Bulletin No 9	First Edition March 2002	EMVB9
[EMV1] + [EMV2] + [EMV3] + [EMV4] + [EMVB6] + [EMVB9]	-	-	EMV
M/Chip 4 Card Application Specification, version 4.0	-	Version 1.0, October 2002	MCAS
M/Chip 4 for MULTOS Version 4.0 Build A Requirements Specification	mxi-mchip4-req-001	3-0	MCMRS
M/Chip 4 Security and Key Management	-	Version 1.0, October 2002	MSKM
Identification cards – Integrated circuit(s) cards with contacts	ISO/IEC 7816	Various	ISO-7816
M/Chip 4 for MULTOS Application Security Target	mxi-mchip-stg-003	4-1	ST

Assumptions

The reader is assumed to have some familiarity with MULTOS and Common Criteria.

Glossary

Term	Definition
AAC	Application Authentication Cryptogram: computed by the application for a declined financial transaction.
AAM	Application Abstract Machine (the interface to MULTOS for applications)
AC	Application Cryptogram
AFL	Application File Locator
AID	Application ID
AIP	Application Interchange Profile
ALU	Application Load Unit
ARPC	Application Response Cryptogram: generated by the Issuer during an on-line transaction and passed to the application using the 2nd GENERATE AC command.
ARQC	Authorisation Request Cryptogram: computed by the application for online card authorisation.
ATC	Application Transaction Counter
CAT3	A CAT3 Terminal has a Terminal Type of 0x26
CBMAC	Card Block MAC
CC	Common Criteria
CDA	Combined DDA/AC
CDOL	Card Risk Management DOL
CFDC	Consecutive Failed Derivation Counter
CHV	CardHolder Verification
CIAC	Card Issuer Action Code
CID	Cryptogram Information Data
CM	Configuration Management
Command	An application command; one of: APPLICATION BLOCK, APPLICATION UNBLOCK, CARD BLOCK, GENERATE AC, GET CHALLENGE, GET DATA, GET PROCESSING OPTIONS, INTERNAL AUTHENTICATE, PIN CHANGE/UNBLOCK, PUT DATA, READ RECORD, UPDATE RECORD, VERIFY. Throughout this document commands are referred to in UPPER CASE.
COTA	Cumulative Offline Transaction Amount
COTN	Consecutive Offline Transaction Number
CVM	Cardholder Verification Method. A CVM list is read by the terminal from the application in order to give the terminal a prioritised list of Cardholder Verification methods. The terminal decides what Cardholder Verification method to use based on this list.
CVR	Card Verification Results
DDA	Dynamic Data Authentication
DDOL	DDA DOL
DES	Data Encryption Standard
DOL	Data Object List
EEPROM	Electrically Erasable Programmable Read Only Memory

Term	Definition
EMV	Europay MasterCard VISA
GP	Grandparent Key used within the EMV2000 session key derivation algorithm
IAD	Issuer Application Data
ICC	Integrated Circuit Card
ICCDN	ICC Dynamic Number
ID	Identifier
LCOTA	Lower Cumulative Offline Transaction Amount
LCOL	Lower Consecutive Offline Limit
M/Chip 4 for MULTOS	The MasterCard EMV application, designed to run on MULTOS, as defined in section 1.7. Referred to within this document as "M/Chip".
MAC	Message Authentication Code
MAL	MULTOS Assembly Language
MEL	MULTOS Executable Language
MK	Master Key
MULTOS	A secure multi-application operating system
N/A	Not Applicable
P	Parent Key used within the EMV2000 session key derivation algorithm
PAN	Primary Account Number
PDOL	Processing Options Data Object List
PIN	Personal Identification Number
POS	Point Of Sale
PVCS	Polytron Version Control System
RSA	Rivest, Shamir, Adleman – a public key algorithm
Script command	A Issuer command, comprising the following 6 commands: • APPLICATION BLOCK • APPLICATION UNBLOCK • CARD BLOCK • PIN CHANGE/UNBLOCK • PUT DATA • UPDATE RECORD
SDA	Static Data Authentication. This is performed by the terminal. The terminal reads static data from the application data in order to perform this. The application does not explicitly perform any functionality to support this and SDA is outside the scope of the evaluation.
SF	Security Function
SFP	Security Function Policy
SFR	Security Functional Requirement
SHA-1	Secure Hash Algorithm
SK	Session Key
SMC	Secure Messaging for Confidentiality

Term	Definition
SMI	Secure Messaging for Integrity
SOF	Strength of Function
SPI	Software Product ID
TBI	To Be Issued
TC	Transaction Certificate: computed by the application for an approved financial transaction.
TOE	Target Of Evaluation
TSC	TOE Scope of Control
TSF	TOE Security Functions
TVR	Terminal Verification Results
UCOTA	Upper Cumulative Offline Transaction Amount
UCOL	Upper Consecutive Offline Limit
Y3	An Authorisation Response Code equal to 0x5933
Z3	An Authorisation Response Code equal to 0x5A33

Conventions

The following conventions are assumed throughout this document:

Term	Definition
∈	Denotes "is a member of"
∉	Denotes "is not a member of"
!=	Denotes "is not equal to"
||	Concatenated
>	Greater than
>=	Greater than or equal to
<	Less than
<=	Less than or equal to
0x	Prefixes a hexadecimal number. For example 0x3A or 0x9901
Byte	8 bits addressable as a single unit
May	Compliance with the requirement or function is *optional*.
Must	Compliance with the requirement or function is *mandatory*.
Shall	Compliance with the requirement or function is *mandatory*.
Should	Compliance with the requirement or function is *highly desirable*.
Word	2 bytes in which the least significant byte has the highest offset.

Table of contents

1 Security Target Introduction 1
1.1 Purpose 1
1.2 ST overview 1
1.3 CC Conformance 1
1.4 Scope 1
1.5 Structure 1
1.6 Document Layout 2
1.7 TOE Reference 2

2 Description of the TOE 3
2.1 Introduction 3
2.2 Intended Use 3
2.2.1 Product Overview 3
2.2.2 Use of M/Chip 3
2.3 Physical scope 6
2.4 Logical scope 7
2.5 Roles 7
2.6 TOE Lifecycle 8

3 TOE Security Environment 9
3.1 Introduction 9
3.2 Assets 9
3.3 Environmental and Method of Use Assumptions 10
3.4 Assumed Threats 11
3.4.1 Masquerade 11
3.4.2 Replay 11
3.4.3 Cloning 11
3.4.4 Disclosure 11
3.4.5 Modification of data 11
3.4.6 False repudiation 11
3.4.7 Denial of service 12
3.5 Organisational Security Policies 12

4 Security Objectives 13
4.1 Security Objectives for the TOE 13
4.2 Security Objectives for the environment 13

5 Security Functional Requirements 15
5.1 TOE Security Functional Requirements summary 15
5.2 Strength of Function 15
5.3 TOE Security Assurance Requirements 15

6 TOE Summary Specification 18
6.1 IT Security Functions 18
6.1.1 SF1 Generate 1st AC 18
6.1.2 SF2 Generate 2nd AC 18

6.1.3 SF3 Dynamic Data Authentication 18
6.1.4 SF4 Key data access 18
6.1.5 SF5 Data integrity 19
6.1.6 SF6 Cardholder authentication 19
6.1.7 SF7 Issuer authentication 19
6.1.8 SF8 Permitted commands 19
6.1.9 SF9 Replay prevention 20
6.1.10 SF10 Issuer Reauthentication 20
6.1.11 SF11 Data management 20
6.1.12 SF12 Transmitted data integrity and confidentiality 20
6.2 Required Security Mechanisms 21
6.3 Assurance Measures 21
6.4 Security functions realised by permutational mechanisms 21

7 Protection Profile Claims 22

1 Security Target Introduction

1.1 Purpose

This document is the summary of the Security Target (see [ST]) used for the Common Criteria evaluation of the M/Chip 4 for MULTOS application.

The role of the security target within the development and evaluation process is described in the Common Criteria for Information Technology Security Evaluation [CC].

1.2 ST overview

The M/Chip 4 for MULTOS application is an EMV compliant credit/debit application. It runs on a smart card platform which is supported by a secure operating system called MULTOS. The scope of this Security Target is the application running on the combined MULTOS and IC platform. A card which contains the M/Chip 4 for MULTOS application may be used in suitable EMV terminals for purchases, linked to a credit or debit account. Transactions may be performed on-line or off-line, or may be declined.

The M/Chip 4 for MULTOS application is generally referred to as "M/Chip" throughout this document.

The requirements for M/Chip 4 for MULTOS may be found in [MCMRS], [MCAS] and [MSKM]. These documents in turn reference the EMV2000 specifications, defined in [EMV].

1.3 CC Conformance

This ST is [CC] Part 2 ([CC1]) conformant and Part 3 ([CC3]) conformant.

The Evaluation Level is EAL4 augmented by AVA_VLA.4, ADV_IMP.2 and ALC_DVS.2.

1.4 Scope

The scope of evaluation is of the M/Chip application only. Note that SDA (Static Data Authentication), as performed purely by the terminal, is outside the scope of this evaluation.

The M/Chip 4 for MULTOS application can be configured for two different application (or "product") profiles, according to the configuration of the application data. These are M/Chip Select 4 and M/Chip Lite 4, the former supporting the entire functionality, the latter omitting RSA-based functions i.e. DDA, encrypted PIN verification and CDA. The Target of Evaluation is the M/Chip 4 for MULTOS application and therefore covers both product profiles that can be data-configured from the single application which forms the TOE.

1.5 Structure

The structure of this document is as defined by [CC] Part 1 Annex C.

- Section 2 is the TOE Description
- Section 3 provides the statement of TOE security environment.
- Section 4 provides the statement of security objectives.

- Section 5 provides the statement of IT security requirements.
- Section 6 provides the TOE summary specification.
- Section 7 provides a statement about protection profile conformance.

1.6 Document Layout

Assumptions are assigned a unique identifier of the form A.<name>, where <name> is a unique label referring to the assumption.

Threats are assigned a unique identifier consistent with CC convention i.e. T.<name>

Security Objectives follow a similar convention i.e. O.<name>, with **Security Objectives for the Environment** being denoted as OE.<name>

Security Functional Requirements (SFRs) use terminology appropriate to the functionality class being as defined in [CC2].

IT Security Functions are denoted as SFn <name>, where n is a positive integer.

Organisational Security Policies are denoted by OSP.<name>.

1.7 TOE Reference

This Security Target is limited to the **M/Chip 4 for MULTOS Application v1.0.1.1**. The application version is determined using the GET DATA command with P1/P2 = 0x9F7E (for Application Life Cycle Data). Bytes 37 and 38 (assuming the first byte returned is byte 0) of the response data will be 0x1010 representing version 1.0.1.1.

2 Description of the TOE

2.1 Introduction

The intent of this section is to describe the TOE as an aid to the understanding of its security requirements, and shall address the product or system type. To this end the scope and boundaries of the TOE are described in general terms both in a physical and a logical way.

2.2 Intended Use

2.2.1 Product Overview

The M/Chip application resides on an ICC which supports a MULTOS Version 4 implementation. MULTOS is a secure multi-application operating system, and hence the M/Chip application may be one of several applications co-residing on the same physical card.

The M/Chip application is an EMV compliant credit/debit application – see [EMV]. The card is used in terminals in order to perform credit or debit transactions and is linked to an account associated with the Cardholder. Transactions may be declined or approved off-line or on-line, where the terminal has on-line capabilities.

2.2.2 Use of M/Chip

The sequence of events during a transaction at a POS payment terminal is defined in the EMV specifications [EMV3], section 6.

Note that the application profile indicates whether M/Chip Select or M/Chip Lite profile is supported by the application. If the M/Chip Lite profile is indicated by the application profile DDA, CDA and encrypted PIN requests are rejected.

In summary the different phases of a typical transaction are as follows:

1. Following selection of the application, the terminal runs GET PROCESSING OPTIONS and reads data (using READ RECORD) based on the response. The terminal then decides, based on the AIP, whether to attempt to perform SDA, DDA or CDA.
2. If SDA is to be performed the terminal reads static data from the application in order to provide authentication of certain application data to the Merchant, on behalf of the Issuer, via the terminal.[1]
3. If required, and supported by the application profile (note that DDA requires RSA functionality, as does CDA [see part 11 below]), the terminal performs DDA using the INTERNAL AUTHENTICATE command. DDA performs dynamic data authentication of the application to the Merchant, acting on behalf of the Issuer, via the terminal. DDA is essentially a challenge-response mechanism providing real-time application authentication.
4. The terminal then decides (based on the CVM list previously read from the application) what form of cardholder verification is to be performed. This might be limited to a pen signature, but may be achieved using the VERIFY command, which may optionally use an encrypted PIN if supported by the application profile (note that encrypted PIN requires RSA functionality).
5. If required and supported by the application profile, cardholder verification may be performed using the encrypted form of the VERIFY command, preceded by the GET CHALLENGE command, if the encrypted option of VERIFY is to be used to protect the PIN. VERIFY performs

[1] The application performs no explicit functionality to support SDA.

authentication of the Cardholder to the application. VERIFY may also be performed using a non-encrypted PIN. In this latter case, a random challenge is not required.

6. The terminal performs the first GENERATE AC requesting the appropriate response. This generates a cryptogram that authenticates the application to the Issuer, indicating offline decline, offline accept or online request.
7. If the transaction is to be performed on-line, the terminal goes on-line to the Issuer system, providing an ARQC which authenticates the application to the Issuer.
8. The Issuer system then provides the ARPC which is passed to the application by the terminal using the 2nd GENERATE AC command. This authenticates the Issuer to the application.
9. Issuer script commands may also be downloaded from the Issuer system to the terminal. These may be performed after the 1st or 2nd GENERATE AC command has returned an AAC or TC.
10. The second GENERATE AC command completes the transaction, with the application generating the response indicating decline or accept. Again this authenticates the application to the Issuer (using the terminal).
11. If supported by the application profile, the cryptogram returned by the 1st or 2nd GENERATE AC command may be wrapped with a RSA signature (known as CDA). CDA requires RSA functionality.
12. Issuer script commands may be performed after the second GENERATE AC command if required. If the transaction is completed during the 1st GENERATE AC command (i.e. it returns an AAC or TC cryptogram), script commands may be performed at this stage.

The communication between the terminal and the ICC is initiated with the activation of the card and the selection of the EMV payment application on the card. This process ensures that the payment AID on the terminal and the payment AID on the card match. Applications are identified by their AIDs (Application Identifier).

Once the application on the card has been selected, the first transaction command normally issued by the terminal is GET PROCESSING OPTIONS. This command initiates the transaction processing in the card and allows a first analysis, to decide if the desired transaction is acceptable from the card's perspective. The application assumes a PDOL length of zero.

As part of the response to the GET PROCESSING OPTIONS the terminal will receive an Application File Locator (AFL). This data is used to locate the data elements on the card's record files, which are required by the terminal for the transaction. The terminal will read all the data elements from locations specified in the AFL using the READ RECORD command.

To verify the authenticity of a card and/or public key data contained in the card data read by the terminal, the terminal will then use either Static Data Authentication (SDA) or Dynamic Data Authentication (DDA), depending on the value of AIP. SDA works just with the data elements read from the card, whereas DDA requires the card to sign some variable data. Both algorithms use asymmetric cryptography to prove the authenticity of data supplied by the application (for SDA) or the authenticity of the application itself (DDA, using the INTERNAL AUTHENTICATE command). The application profile may or may not support DDA or CDA, depending on data configuration.

The terminal will employ SDA, which allows the Merchant to authenticate certain application data on behalf of the Issuer, by verifying the authenticity of certain data relating to the M/Chip Application. SDA is the off-card verification of a (static) digital signature. SDA cannot defend against replay of this Issuer-signed data. Successful SDA verification allows the terminal to confirm that there is a genuine account, with *bona fide* account details as recovered. Optionally DDA allows the Merchant, acting via the terminal on behalf of the Issuer, to recover and check a card level certified public key. The terminal uses the INTERNAL AUTHENTICATE, if supported by the application profile, to perform a challenge and response protocol to determine that the application is not an unauthorised copy. The DDA, if required, is performed off-line in real time without connection to the Issuer system.

Next in sequence is cardholder verification. Cardholder verification by the card consists of the verification of a Personal Identification Number (PIN). A cardholder is requested to enter a PIN into

the terminal PIN Pad, and this number is passed to the card in the VERIFY command - for comparison with an internally stored reference PIN value. To ensure a safe transmission of the PIN from the terminal into the ICC, asymmetric encryption using a random number can be used if supported by the application profile. This random number is retrieved from the card with the GET CHALLENGE command, immediately prior to performing the VERIFY command.

The terminal will then perform terminal risk management and terminal action analysis. During these processes, the terminal will make a first decision on whether it considers that the transaction should be completed off-line (accepted or declined), or if on-line approval by the issuer is required. Based on that decision, the terminal will issue the GENERATE AC command to the card, requesting either an off-line (TC or AAC) or on-line completion of the transaction (ARQC). The GENERATE AC command may, if requested and supported by the application profile, generate a Combined DDA/AC signature (CDA), which wraps the DES-based Application Cryptogram within an RSA signature (only for TC or ARQC responses).

At this stage, the card will perform its own card risk management based on transaction parameters and information gathered during previous transactions. The card will accept the request for off-line processing, or decline the transaction, or it will request the terminal to call the issuer for approval. If the transaction can be completed off-line, then the card will update its internal data and pass a positive response to the terminal (TC). In this case, the transaction processing between the terminal and the ICC is complete.

If the terminal receives from the card a request to go on-line (ARQC), either because it requested such processing in the first GENERATE AC command or because the card made this decision during card action analysis, then it will send an authorisation request message to the issuer's authorisation system. To allow the issuer to verify the authenticity of the request, the card generates a symmetric cryptogram during the first GENERATE AC command (ARQC). In return, the authorisation system will itself send data to the terminal for the card, which it protects with a symmetric cryptogram (ARPC).

In addition to the response to the authorisation request, the issuer may choose to include additional commands for the card. These issuer script commands are transmitted to the card by the terminal as part of the issuer-to-card script processing, after the second GENERATE AC command. These commands perform management tasks such as application blocking or unblocking, card blocking (at the MULTOS level), PIN change or unblocking and management of security attributes to be performed. Script commands may only be performed after an AAC or TC has been returned after the (first or second) GENERATE AC command.

Once the terminal receives an authorisation response from the issuer, it may use the second GENERATE AC command to allow the card to authenticate the issuer authorisation system.

The terminal will then complete the transaction with a second GENERATE AC command. On this iteration, the card will again decide based on information gathered during the previous steps, if it can approve (TC) or decline (AAC) the transaction.

Once the terminal receives the last response from the card, the transaction is complete, unless any subsequent Issuer script commands are to be performed.

2.3 Physical scope

The application resides on an ICC which supports MULTOS version 4. This will normally be in the form of credit card sized card. The card communicates with the terminal via the MULTOS I/O interface, which is [ISO-7816] compliant. Communication between the terminal and Issuer system is outside the scope of this Security Target. MULTOS applications perform MULTOS instructions and primitives via the MULTOS Application Abstract Machine.

The scope of the evaluation is the application only – the MULTOS platform is not included in the application and the application evaluation is independent of any specific MULTOS 4 platform.

The terminal may have on-line capability or may be off-line only. In addition the terminal may provide a means for the PIN to be entered by the Cardholder and will provide a means of displaying the value of the transaction to the Cardholder.

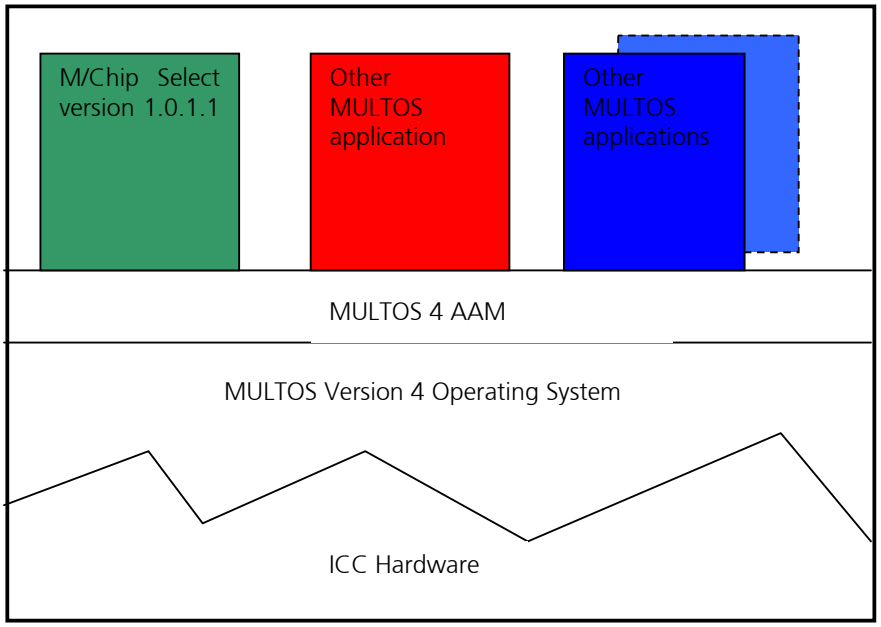

Figure 1 Logical representation of ICC containing MULTOS and applications

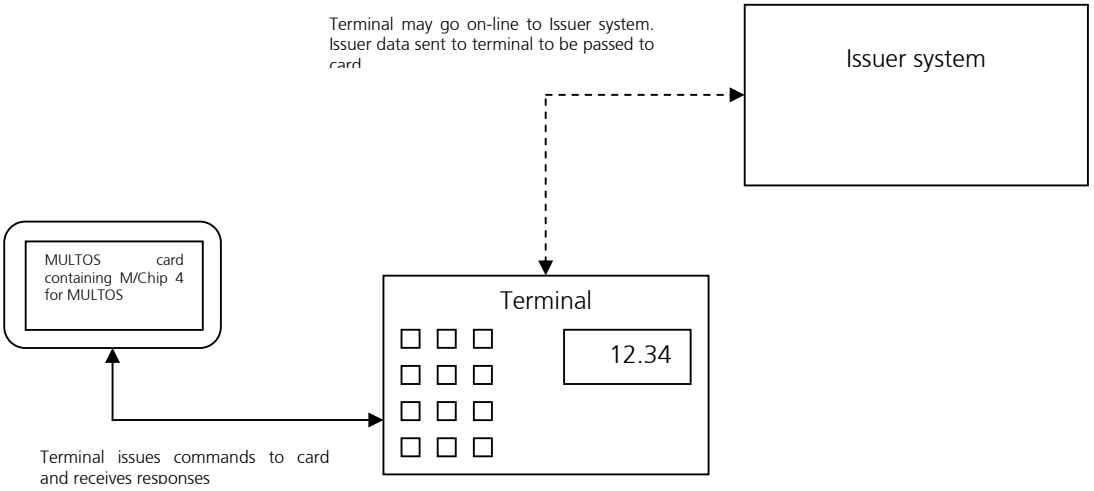

Figure 2 Main system elements

2.4 Logical scope

The scope of the TOE is the M/Chip application.

Note also that functionality concerning SDA is not within scope, as this is performed purely by the terminal, which reads static data from the application.

2.5 Roles

There are only two roles considered within this Security Target:

- Cardholder – the legitimate holder and user of the card/application. The application is linked to an account controlled by the Cardholder. The Cardholder owns the PIN for the application.

- Issuer – a bank responsible for issuing and managing cards containing the M/Chip application.

Note that the terminal is not considered as a role for the purposes of this security target; it acts as the interface between the card and the Issuer and also the card and the Cardholder.

Note also that Cardholders, Acquirers, Merchants and Issuers are all bound by a set of scheme rules which dictate the terms of acceptance and processing of card transactions. Therefore the Merchant and Acquirer are considered as agents acting on behalf of the Issuer.

2.6 TOE Lifecycle

The following diagram gives an overview of the TOE lifecycle. The TOE development environment is limited to the application development by Mondex International Ltd (MXI) based at its London offices which are located at 47 – 53 Cannon Street, London EC4M 5SH, UK. Once developed the application is securely delivered to the Mondex Customer Care Team who are located at the Mondex offices at St. Andrews House, The Links, Kelvin Close, Birchwood, Warrington, Cheshire, WA3 7PB, UK.

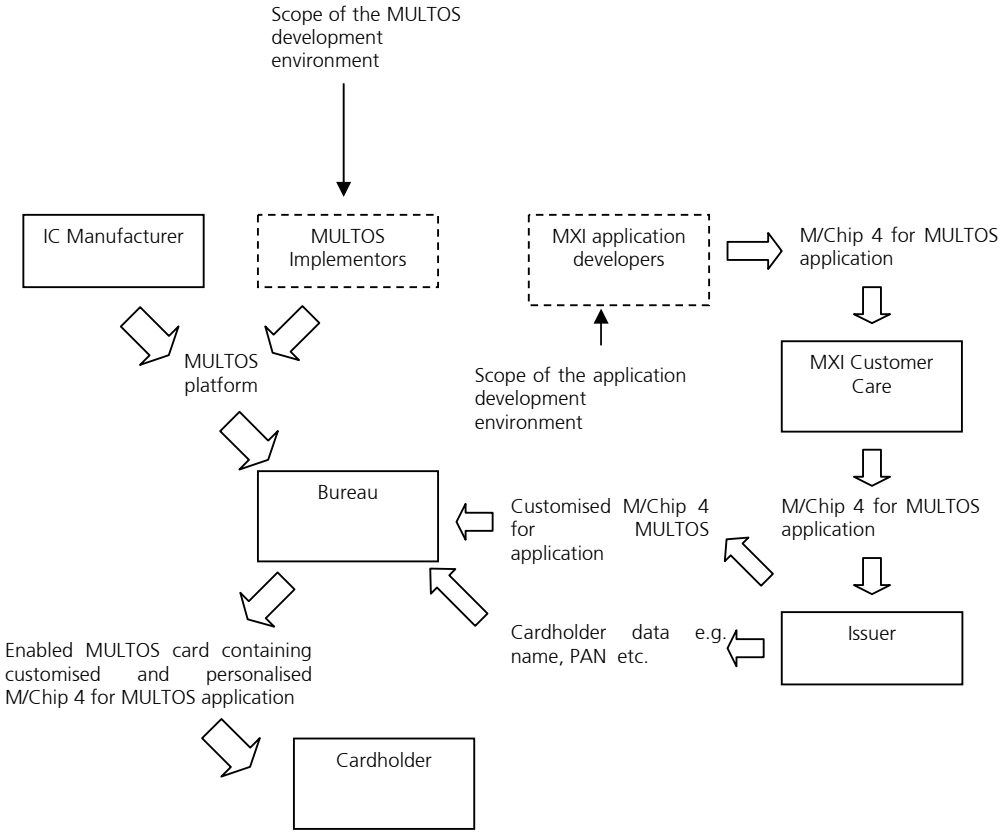

Figure 3 TOE Lifecycle overview

3 TOE Security Environment

3.1 Introduction

The statement of TOE security environment describes the security aspects of the environment in which the TOE is intended to be used and the manner in which it is expected to be employed.

To this end, the statement of TOE security environment identifies the assumptions made on the operational environment (including physical and procedural measures) and the intended method of use of the product. It also defines the threats that the application is designed to counter, and the organisational security policies with which the application is designed to comply.

3.2 Assets

Assets are security relevant elements of the TOE or used by the TOE.

- The primary assets for the TOE are the secret cryptographic keys and PIN data stored within the TOE:
  - Reference PIN Block
  - ICC Private Key
  - ICC PIN Encipherment Private Key
  - MK_{IDN}
  - MK_{AC} (for EPI/MCI session key derivation algorithm only)
  - MK_{SMI} (for EPI/MCI session key derivation algorithm only)
  - MK_{SMC} (for EPI/MCI session key derivation algorithm only)
  - GP_{AC}, P_{AC} (for EMV 2000 session key derivation algorithm only)
  - GP_{SMI}, P_{SMI} (for EMV 2000 session key derivation algorithm only)
  - GP_{SMC}, P_{SMC} (for EMV 2000 session key derivation algorithm only)
  - Session keys

- The secondary assets for the TOE are the following application and transaction data used by the TOE to perform its functions. The secondary assets are defined as:
  - AFL
  - AIP
  - Application Control
  - Application Currency Code
  - ARPC Response Code
  - ATC
  - Card Issuer Action Codes (Decline, Default and Online)
  - Cardholder data (PAN)
  - CDOL1 and CDOL2
  - CVM List
  - CVR
  - Issuer Velocity Params (LCOTA, UCOTA, LCOL, UCOL, Currency Conversion Table)
  - Offline Terminal Types
  - PIN Try Counter
  - PIN Try Limit

- Script MAC Counter
- SDA Tag List
- Velocity Params (COTA, COTN)

The primary assets are to be protected in terms of confidentiality and integrity.

The secondary assets are to be protected in terms of integrity.

Note that assets are protected as follows:

- *Logically, by the application – the subject of this security target*
- *Logically, by the platform – see A.DOMAIN*
- *Physically, by the IC – see A.TAMPER*

User Data is defined as secret key data (both symmetric and asymmetric).

3.3 Environmental and Method of Use Assumptions

Assets are to be protected by the TOE itself. However, it must be combined with some technical, physical and/or organisational countermeasures to be enforced within the TOE environment.

Thus, the following general assumptions are made upon the TOE security environment:

A.CHV_INSTALL	The PIN must be delivered to the Cardholder via secure means, that are separate from the delivery of the card itself.
A.SEC_LOAD	It is assumed that cryptographic key and PIN data has its confidentiality maintained throughout the processes of generation and loading into the application data prior to loading onto the card.
A.TAMPER	It is assumed that platform (chip and operating system) on which the TOE runs is resistant to physical attack such that sensitive application data cannot be read, modified or inferred using physical attack or monitoring of power and/or timing data during execution of the application.
A.DOMAIN	It is assumed that the platform (chip and operating system) on which the TOE runs prevents other applications that have been loaded from accessing (reading or writing) the code and data of the TOE. It is also assumed that the operating system logically protects the application code and data from unauthorised modification and disclosure. Additionally it is assumed that the platform correctly executes the MEL instructions and MULTOS primitives that comprise the TOE.

3.4 Assumed Threats

This section will define the assumed threats to the TOE. For each threat it is assumed that the attacker will have a high attack potential, in terms of motivation, expertise and available resources.

3.4.1 Masquerade

T.USRP_USR	An attacker may masquerade as the Cardholder, in order to perform unauthorised Cardholder functions.
T.USRP_ISS	An attacker may masquerade as the Issuer, in order to perform unauthorised Issuer functions.

3.4.2 Replay

T.REP_CHV	An attacker may perform a replay attack on Cardholder verification.
T.REP_ISS	An attacker may perform a replay attack on Issuer Script commands.
T.REP_APPL	At attacker may perform a replay attack on data that authenticates the application.

3.4.3 Cloning

T.CLON_APP	An attacker may clone application and/or key data, leading to forgery of transaction and authentication data.

3.4.4 Disclosure

T.DIS_KEY	An attacker may perform attacks leading to the disclosure of key data.
T.DIS_CHV	An attacker may perform attacks leading to the disclosure of Cardholder verification data.

3.4.5 Modification of data

T.MOD_KEY	An attacker may perform attacks leading to the modification of key data.
T.MOD_CHV	An attacker may perform attacks leading to the modification of Cardholder verification data.
T.MOD_ATTR	An attacker may perform attacks leading to the unauthorised modification of TOE application data and transaction data.

3.4.6 False repudiation

T.FAL_REP	An actor[2] in the system may falsely repudiate a transaction.

[2] An actor is defined as the Cardholder or the Issuer (or a person acting on behalf of the Issuer).

3.4.7 Denial of service

T.DEN_SER	An attacker may perform a denial of service attack.

3.5 Organisational Security Policies

The following rules concern the organisational security policies relating to the TOE:

OSP.CH_BEH	The cardholder shall keep their PIN secret and shall not lend their card to other persons.
OSP.INTENT	Each transaction is an intentional operation of the Cardholder. A procedure defined by the Issuer shall exist in order to allow the Cardholder to either accept or decline the transaction.
OSP.IDENT	Each card shall uniquely identify the account to which it is linked.
OSP.SDA	Data stored in and readable from the application shall enable static off-line authentication of some application data by a terminal.
OSP.VAL_IND	There shall exist a means to indicate to the Cardholder the amount of the proposed transaction.
OSP.MANAGE	The Issuer shall manage the TOE such that IT security is maintained. In particular sensitive data (e.g. key data, PIN data and security attributes) must be protected. Administrators of the system must be vetted and trained appropriately.
OSP.ATC	The Issuer system shall monitor ATC usage in cryptograms to ensure that cards have not been cloned.
OSP.ONLINE	Terminals shall be instructed to perform all transactions online if it is not possible to perform DDA.

4 Security Objectives

4.1 Security Objectives for the TOE

The M/Chip application will conform to the following security objectives:

O.AUTH_USR	When requested, the TOE shall authenticate the Cardholder.
O.AUTH_ISS	The TOE shall authenticate the Issuer prior to performing Issuer authorised functions.
O.REPLAY	The TOE shall prevent replay attacks on the following: • Encrypted PIN data • Issuer Script commands • DDA data • Application Cryptogram and CDA data
O.CLONE	The TOE shall prevent determination of key data, in order to prevent forgery of transactions, authentication and scripts.
O.DISCLOSE	The TOE shall prevent disclosure of primary assets
O.MOD_DAT	The TOE shall prevent unauthorised modification of primary and secondary assets
O.REPUD	The TOE shall ensure that transaction and authentication data cannot be falsely repudiated at a later time
O.AVAIL	The TOE shall ensure service availability.

4.2 Security Objectives for the environment

OE.TERMINAL	The terminal must support the relevant EMV functionality and be appropriately tested and approved. The terminal communicates with the application, via the MULTOS platform, using a command/response communications protocol in accordance with [ISO-7816]. The terminal must be able to support both static and dynamic data authentication of the card and also provide an indication of the amount of the transaction to the cardholder. The terminal shall ensure that Cardholder Verification data is kept confidential at all times and securely deleted as soon as it is no longer required.
OE.SYSTEM	The Issuer and Cardholder shall apply the system security policy. The Issuer shall communicate to the Cardholder the relevant rules for using the TOE.
OE.INSTALL	The Issuer shall ensure that the TOE is delivered and installed in a manner that maintains IT security, including unique identification of the account to which the card is linked and SDA data.
OE.MANAGE	The Issuer shall ensure that the TOE is managed, administered and operated in a manner that maintains IT security.
OE.TAMPER	The platform that the application runs on shall be resistant to physical attack such that sensitive application data cannot be read, modified or inferred

	using physical attack or monitoring of power and/or timing data during execution of the application.
OE.DOMAIN	The platform that the application runs on shall maintain a separate domain from other applications for the M/Chip application. The operating system logically protects the application code and data from unauthorised modification and disclosure. The platform correctly executes the MEL instructions that comprise the application.
OE.BLOCK	The MULTOS platform on which the TOE runs shall provide a Card Block primitive.
OE.DATA_SEC	All TOE related data stored off-card must be protected in order to maintain appropriate security. For sensitive data this requires protection of confidentiality from generation through load into the application data and maintenance thereafter.
OE.ADM_SEC	Personnel working as administrators shall be carefully selected for reliability and appropriately trained.
OE.INTEGRITY	The MULTOS platform on which the TOE runs shall check the integrity of the application code upon application selection.
OE.CHV_INSTALL	The PIN shall be securely delivered to the cardholder, separate from the card.
OE.FCS_COP	The MULTOS platform shall provide DES and RSA primitives for the application; the platform will ensure that the primitives execute correctly and do not leak key information.
OE.ATC	The Issuer system will determine, through monitoring of valid cryptograms, that: • The same ATC is not used for the same card and for different transactions and • The usage of the ATC in cryptograms for a given card is consistent. Violations of these rules will indicate that the card has most likely been cloned.
OE.ONLINE	Terminals will perform transactions online if it is not possible to perform DDA.

5 Security Functional Requirements

5.1 TOE Security Functional Requirements summary

The following TOE Security Functional Requirements were used in [ST] :

- FCO_NRO.2 : Enforced proof of origin
- FDP_ACC.2 : Complete Access Control
- FDP_ACF.1 : Security attribute based access control
- FDP_IFC.1 : Subset Information flow control
- FDP_IFF.1 : Simple Security Attributes
- FDP_SDI.2 : Stored data Integrity monitoring and action
- FIA_AFL.1 : Authenticaiton failure handling
- FIA_UAU.1 : Timing of authentication
- FIA_UAU.4 : Single-use authentication mechanisms
- FIA_UAU.6 : Re-authenticating
- FMT_MTD.1 : Management of TSF data
- FMT_SMR.1 : Security roles
- FTP_TRP.1 : Trusted path

The following Security Functional Requirements for the TOE IT Environment were used in [ST] :

- FCS_COP.1 : Cryptographic Operation

5.2 Strength of Function

All security mechanisms in [ST] have a claimed minimum Strength of Function of *SOF-High* [CC] for all security mechanisms.

For cryptographic mechanisms, the relevant algorithms are publicly known and as such no further comment will be made on their suitability or strength.

5.3 TOE Security Assurance Requirements

The target evaluation assurance level for the product is EAL4 [CC]. Augmented assurance requirements are AVA_VLA.4, ADV_IMP.2 and ALC_DVS.2.

6 TOE Summary Specification

6.1 IT Security Functions

6.1.1 SF1 Generate 1st AC

6.1.1.1 SF1.1 Initial Checks

This SF defines the initial checks performed by the application during the 1st GENERATE AC command.

6.1.1.2 SF1.2 Determine AC type

This SF defines how the AC type is determined for the 1st GENERATE AC command.

6.1.1.3 SF1.3 Construct AC/CDA

This SF defines how the AC (and optionally CDA) is generated for the 1st GENERATE AC command.

6.1.2 SF2 Generate 2nd AC

6.1.2.1 SF2.1 Initial Checks

This SF defines the initial checks performed by the application during the 2nd GENERATE AC command.

6.1.2.2 SF2.2 Determine AC type

This SF defines how the AC type is determined for the 2nd GENERATE AC command.

6.1.2.3 SF2.3 Construct AC/CDA

This SF defines how the AC (and optionally CDA) is generated for the 2nd GENERATE AC command.

6.1.3 SF3 Dynamic Data Authentication

This SF defines how signatures are generated for DDA.

6.1.4 SF4 Key data access

6.1.4.1 SF4.1 RSA Keys

This SF defines the rules for access to RSA keys within the application.

6.1.4.2 SF4.2 MK_{IDN}

This SF defines the rules for access to the MK_{IDN} key within the application.

6.1.4.3 SF4.3 EPI/MCI Master Keys

This SF defines the rules for access to EPI/MCI Master Keys within the application.

6.1.4.4 SF4.4 EMV2000 keys

This SF defines the rules for access to EMV2000 Keys within the application.

6.1.4.5 SF4.5 Session keys

This SF defines the rules for access to Session Keys within the application.

6.1.4.6 SF4.6 Additional details

This SF defines additional rules for key access.

6.1.5 SF5 Data integrity

This SF defines how data integrity is protected by the application.

6.1.6 SF6 Cardholder authentication

6.1.6.1 SF6.1 Actions taken if VERIFY command fails

This SF defines the actions taken by the application if Cardholder authentication fails.

6.1.6.2 SF6.2 Resetting the PIN Try Counter

This SF defines how the PIN Try Counter may be reset.

6.1.7 SF7 Issuer authentication

6.1.7.1 SF7.1 Issuer Authentication fails during a script command

This SF defines the actions taken by the application if Issuer authentication fails during a script command.

6.1.7.2 SF7.2 Issuer Authentication fails during the 2nd GENERATE AC command

This SF defines the actions taken by the application if Issuer authentication fails during the 2nd GENERATE AC command.

6.1.8 SF8 Permitted commands

6.1.8.1 SF8.1 Commands that require Cardholder or Issuer Authentication

This SF defines the commands that require Cardholder or Issuer Authentication.

6.1.8.2 SF8.2 Issuer Authentication during the 2nd GENERATE AC command

This SF defines how the application may authenticate the Issuer during the 2nd GENERATE AC command.

6.1.8.3 SF8.3 Use of the Command Table

This SF defines how command execution is governed by the Command Table.

6.1.9 SF9 Replay prevention

6.1.9.1 SF9.1 GENERATE AC command cryptograms

This SF defines how GENERATE AC command cryptograms protect against replay.

6.1.9.2 SF9.2 DDA signatures

This SF defines how DDA signatures protect against replay.

6.1.9.3 SF9.3 Issuer script commands

This SF defines how Issuer Script commands protect against replay.

6.1.9.4 SF9.4 CARD BLOCK command

This SF defines how the CARD BLOCK command protects against replay.

6.1.9.5 SF9.5 VERIFY command

This SF describes VERIFY command protects against replay.

6.1.10 SF10 Issuer Reauthentication

This SF defines the action to be taken if Issuer Authentication fails during any script command.

6.1.11 SF11 Data management

6.1.11.1 SF11.1 PUT DATA and UPDATE RECORD commands

This SF defines the rules for the updating of data within the application using the PUT DATA and UPDATE RECORD commands.

6.1.11.2 SF11.2 PIN CHANGE/UNBLOCK command

This SF defines the rules for the updating of PIN-related data within the application using the PUT DATA and UPDATE RECORD commands.

6.1.11.3 SF11.3 Checksum updates

This SF defines how checksum are updated within the application.

6.1.11.4 SF11.4 Sensitive data check

This SF defines additional rules for the handing of sensitive data.

6.1.12 SF12 Transmitted data integrity and confidentiality

6.1.12.1 SF12.1 New PIN

This SF defines how the integrity and confidentiality of new PIN data supplied via a script command is maintained.

6.1.12.2 SF12.2 PIN

This SF defines how the integrity and confidentiality of PIN data supplied via the encrypted form of the VERIFY command is maintained.

6.1.12.3 SF12.3 CBMAC

This SF defines how the integrity and confidentiality of CARD BLOCK data supplied via the CARD BLOCK command is maintained.

6.2 Required Security Mechanisms

There are no requirements for specific security mechanisms.

6.3 Assurance Measures

Appropriate assurance measures will be adopted to address each of the EAL4 assurance requirements augmented by AVA_VLA.4, ALC_DVS.2 and ADV_IMP.2. See section 0.

6.4 Security functions realised by permutational mechanisms

The following Security Functions are realised by permutational mechanisms (these include Security Functions that are realised by cryptographic mechanisms):

- SF1
- SF2
- SF3
- SF6
- SF8
- SF9
- SF10
- SF11
- SF12

The Strength of Function for each of SF1, SF2, SF3, SF6, SF8, SF9, SF10, SF11 and SF12 is SOF-High.

No other Security Functions are realised by a permutational or probabilistic or cryptographic mechanism.

7 Protection Profile Claims

The Security Target in [ST] makes no claims about conformance with any protection profile.

** End of document ***