The James Watt Building,
Scottish Enterprise Technology Park,
East Kilbride
Scotland, G75 OQD
United Kingdom

Ecebs ISAM/MFOS (Multefile) Security Target Lite

Release Date: 31-oct-2005

Version: 6.5 Lite

Document Reference: ITSO-STR-002-L2

Change History

Version	Name	Description	Date
6.5		Updates to TOE version & references.	20-Apr-2005
6.5 Lite		Creation of publishable version	31-Oct-2005

Table of Contents

1 ST Introduction

1.1 ST Identification

Title: **Ecebs ISAM/MFOS (Multefile) Security Target.**

ST Version: **6.5 Lite**

Components: **Ecebs MFOS (Multefile) operating system and ISAM application.**

Atmel 3232CS smartcard device.

ATMEL AT45DB321B 32-Mbit Flash memory device.

TOE Version: **00_06_13**

A glossary of the terms used is given in Annex A.
This Security Target has been constructed with 1.6 "Protection Profile - Smart Card Integrated Circuit with Embedded Software", Version 2.0, Issue June 1999, registered at the French Certification Body under the number PP/9911.

1.2 ST Overview

This Security Target covers the development and the active phases of an Integrated Transport Smartcard Organisation Secure Application Module (ISAM).
The ISAM is able to receive and manage transport application data. The ISAM supports the provision of interoperable contactless smartcard public transport ticketing services (ITSO Shell) in a manner, which offers end-to-end loss-less data transmission and security.

The purpose of the ISAM is:

- To provide functionality to support authentication of ITSO Shells and identification of ITSO terminals.
- To provide functionality to support calculating unique keys to access ITSO Shells.
- To provide functionality to support verification of data integrity.
- To provide functionality to support certification of data modification.
- To provide functionality to support secure storage and updating of ISAM capabilities.
- To provide functionality to support secure storage, recovery and verification of system transactions for the purpose of clearing and settlement.
- To provide functionality to support secure loading and storage of “Hot Lists” and “Action Lists”.
- To provide functionality to support the prevention of incorrect command operation and sequencing.

1.3 CC conformance

This Security Target is conformant to:

- 1.6"Common Criteria for information Technology Security Evaluation, Part 2: Security Functional Requirements", August 1999, version 2.1, CCIMB-99-032
- 1.6"Common Criteria for information Technology Security Evaluation, Part 3: Security Assurance requirements", August 1999, version 2.1, CCIMB-99-033

As follows:

- Part 2 conformant: the security functional requirements are based only upon functional components identified in part 2 of the Common Criteria.
- Part 3 conformant: the security assurance requirements are based only upon assurance components in part 3 of the Common Criteria.

The ST is conformant to a pre-defined named assurance package as follows:

- EAL 4 augmented: the security assurance requirements are a proper superset of all assurance components in EAL 4.
- The augmentation relates to the requirement to meet the following components AVA_VLA.4 Highly resistant, ALC_DVS.2 Sufficiency of security measures and ADV_IMP.2 Implementation of the TSF.
- The strength level for the TOE Security functions is “SOF-high” (Strength of Functions High).

The ST is conformant to a Protection Profile as follows:

- 1.6"Protection Profile - Smart Card Integrated Circuit with Embedded Software", Version 2.0, Issue June 1999, registered at the French Certification Body under the number PP/9911

1.4 Document Objectives

The purpose of this document is to satisfy the Common Criteria requirements for a Security Target for the ISAM, which utilises the Ecebs Multi-Function Operating System (MFOS) and Ecebs secure application management technology (Multefile) for smart card operating systems.

1.5 Document Structure

Chapter 1 introduces the Security Target.

Chapter 2 provides a description of the TOE, as an aid to the understanding of its security requirements, and addresses the product type, the intended usage, and the general features of the TOE.

Chapter 3 describes the TOE security environment.

Chapter 4 describes the required security objectives for the TOE and its environment.

Chapter 5 describes the TOE Security functional and assurance requirements and the security requirements for the TOE's IT environment.

Chapter 6 describes the TOE security functions, which satisfy the previously stated TOE security functional requirements and the assurance measures, which satisfy the TOE security assurance requirements.

Chapter 7 describes the PP claims.

Chapter 8 describes the security objectives rationale, security requirements rationale, TOE summary specification rationale and PP claims rationale.

1.6 References

[1] "Common Criteria for information Technology Security Evaluation, Part 1: Introduction and general model", August 1999, version 2.1, CCIMB-99-031

[2] "Common Criteria for information Technology Security Evaluation, Part 2: Security Functional Requirements", August 1999, version 2.1, CCIMB-99-032

[3] "Common Criteria for information Technology Security Evaluation, Part 3: Security Assurance requirements", August 1999, version 2.1, CCIMB-99-033

[4] "Protection Profile - Smart Card Integrated Circuit with Embedded Software", Version 2.0, Issue June 1999, registered at the French Certification Body under the number PP/9911

[5] "FIPS PUB 46-3, Data Encryption Standard", October 25, 1999 (ANSI X3.92), National Institute of Standards and Technology

[6] " FIPS PUB 81 DES Modes of Operation", December 2, 1980 (inc. Change Notices 2, 3)

[7] FIPS PUB 180-1, Secure Hash Standard", April 17, 1995, National Institute of Standards and Technology

[8] "Information technology – Security techniques: Message authentication codes (MACs)—Part 1: Mechanisms using a block cipher", ISO 9797-1 (1999)

[9] "PKCS#1 v2.0: RSA Encryption Standard, RSA Laboratories", October 1998

[10] "Atmel AT90SC3232CS Data Sheet" 22 September 2003

[11] "Protection Profile, Smartcard Integrated Circuit", Version 2.0, Issue September 1998 PP/9806

[12] "FIPS PUB 140-1, Security Requirements for Cryptographic Modules", Jan 1994

[13] Project ITSO, Interface Control Document, ITSO-ICD-001-L3E, Revision 6.2, 26th October 2004.

[14] ISAM TOE Security Policy Model, ITSO-TSPM-001-L3E, Version 6.3, 13th April 2005.

[15] Atmel AT90SC3232CS Security Target Lite, Revision Date, 30th April 2004.

[16] ISO/IEC 7816-3, Identification Cards- Integrated circuit(s) cards with contacts- Part 3: Electronic Signals and Transmission Protocols.

[17] ISO/IEC 7816-4, Identification Cards- Integrated circuit(s) cards with contacts- Part 4: Interindustry Commands for Interchange.

2 TOE Description

This part of the ST describes the TOE as an aid to the understanding of its security requirements and addresses the product type, the intended usage and the general features of the TOE. The scope and boundaries of the TOE shall also be described both in physical terms (hardware and/or software components/modules) and logical terms (IT and security features offered by the TOE).

2.1 Product Type

The Target of Evaluation (TOE) is the Smart Card Integrated Circuit with Embedded Software in operation. The TOE is independent of the physical interface, the way it is packaged and any other security device supported by the physical card base. Specifically, the TOE consists of the ISAM application and the MFOS operating system residing on a smartcard module comprising an Atmel 3232CS smartcard device. The module will also include an ATMEL AT45DB321B 32-Mbit Flash memory device, which is outside the scope of the TOE. However, any confidential data stored in the External Memory (XMEM) shall be cryptographically protected.

The **Atmel 3232CS** smartcard device is made up of a number of hardware modules including a processing unit, security components, I/Os and volatile and non-volatile memories as per the 1.6"Atmel AT90SC3232CS Data Sheet" 22 September 2003 The Atmel 3232CS shall be common criteria certified as conformant to 1.6 "Protection Profile, Smartcard Integrated Circuit", Version 2.0, Issue September 1998 PP/9806

2.1.1 Logical Modules

The TOE (ISAM) is composed of the following subsystems identified in the diagram below:

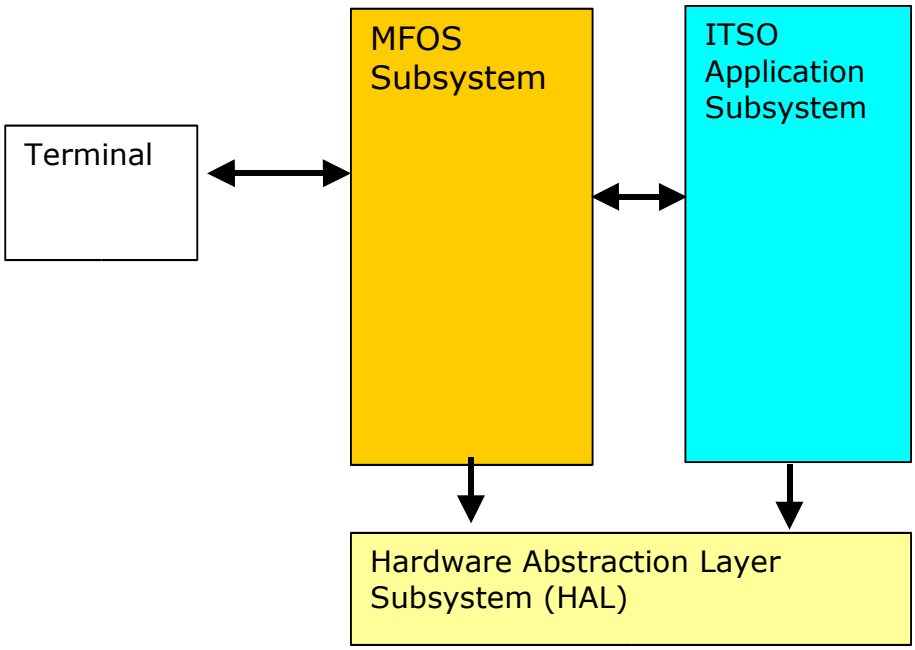

Figure 1 ISAM logical modules

The system interfaces can be seen in Figure 1 ISAM logical modules.

Interfaces are provided by the MFOS(Multefile) Subsystem (File System and Personalisation) to support the reception of external Application Protocol Data Units (APDUs) from the Terminal and also internal function calls from the ITSO Application Subsystem (including the security environments).

Interfaces are provided by the ITSO Application Subsystem to support the reception of internal function calls from the MFOS Subsystem.

Interfaces are provided by the HAL (Hardware Abstraction Layer) Subsystem to support the reception of internal function calls from the ITSO Application Subsystem and the MFOS subsystem.

The Terminal is not part of the TOE.

2.2 Smart Card Product Lifecycle

The Smart Card Product lifecycle is de-composed in 7 phases, according to 1.6"Protection Profile - Smart Card Integrated Circuit with Embedded Software", Version 2.0, Issue June 1999, registered at the French Certification Body under the number PP/9911 (Figure 2).

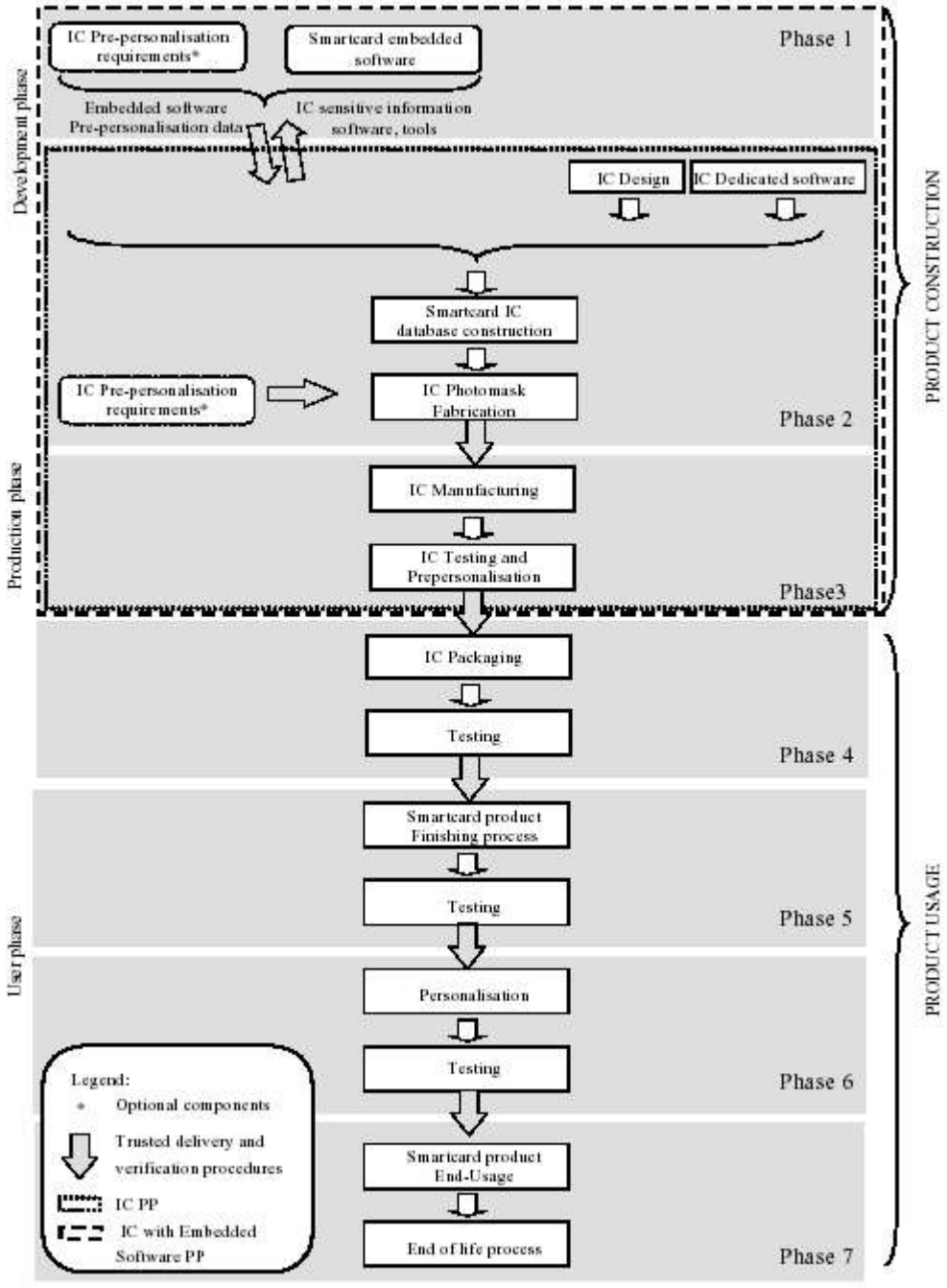

Figure 2

The following authorities are involved in the 7 phases identified in the above diagram:

Phase 1	Smartcard software development	The **smartcard software developer** is in charge of the Basic Software and the Application Software development and the specification of initialisation requirements.
Phase 2	IC development	The **IC designer** designs the IC, develops the IC dedicated software, provides information, software or tools to the smartcard software developer, and receives the software from the developer, through trusted delivery and verification procedures. From the IC design, IC dedicated software and smartcard embedded software, the IC designer constructs the smartcard IC database, necessary for the IC photomask fabrication. The **smartcard software developer** is responsible for the ICC Pre-personalisation requirements.
Phase 3	IC manufacturing and testing	The **IC manufacturer** is responsible for producing the IC through three main steps: IC manufacturing, IC testing, and IC pre-personalisation.
Phase 4	IC packaging and testing	The **IC packaging manufacturer** is responsible for the IC packaging and testing.
Phase 5	Smartcard product finishing process	The **smartcard product manufacturer** is responsible for the smartcard product finishing process and testing.
Phase 6	Smartcard personalisation	The **personaliser** is responsible for the smartcard personalisation and final tests. Other application software may be loaded onto the chip at the personalisation process.
Phase 7	Smartcard end-usage	The **smartcard issuer** is responsible for the smartcard product delivery to the **smartcard end-user**, and the end of life process.

Figure 3

The Embedded Software designed during Phase 1 controls and protects the TOE during Phases 4 to 7 (Product Usage). The limits of this Development Environment correspond to Phase 1 (including the delivery and verification procedures and the TOE delivery to the IC Designer) and Phases 2 and 3 (in conjunction with 1.6"Protection Profile, Smartcard Integrated Circuit", Version 2.0, Issue September 1998 PP/9806).

2.3 TOE Environment

The TOE environment is defined as follows:

- Development environment (corresponding to Phase 1 and including the relevant pre-personalisation requirements),
- IC Development and Photomask Fabrication environment (corresponding to phase 2 and addressed by the Smart Card IC PP 9806/ v2.0 for the Atmel 3232CS part),
- IC manufacturing environment corresponding to phase 3, including the integration of the TOE in the IC and the test operations,
- IC Packaging, and Smart Card Product Finishing process environment (corresponding to phases 4 and 5), including test operations,
- Personalisation environment corresponding to personalisation and testing of the Smart Card with the user data (phase 6) and,
- End-User environment (phase 7).

2.3.1 TOE Development Environment

Phase 1:
To assure security, the environment in which the development takes place is made secure with controllable access. All authorised personnel involved fully understand the importance of the rigid implementation of defined security procedures.

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

Design and development of the TOE then follows. The engineer uses a secure computer system (preventing unauthorised access) to make his design, implementation and test performances.

Sensitive documents, databases on tapes, disks and diskettes are stored in appropriately locked cupboards and safes. Disposal of unwanted data is carried out by shredding (paper documents) or complete electronic erasures (electronic documents, databases).

Testing, programming and deliveries of the TOE then take place.

During offsite deliveries of the TOE, the TOE is transported according to prescribed delivery processes

2.3.2 TOE Production Environment

Phases 2 and 3:
This production environment for the Atmel 3232CS part is defined in1.6"Protection Profile, Smartcard Integrated Circuit", Version 2.0, Issue September 1998 PP/9806

2.3.3 TOE User Environment

Phases 4 and 5:
During phases 4 and 5 of production, the TOE is used in the IC Packaging, Smart Card Finishing process and the test environments. All authorised personnel involved in these operations fully understand the importance of the rigid implementation of defined security procedures.
The environment in which these operations take place is appropriately secured. Sensitive information (tapes, disks or diskettes) is stored in appropriately locked cupboards and safes. Disposal of unwanted data is carried out by shredding (paper documents) or complete electronic erasures (electronic documents, databases).

Phase 6:
Established control procedures shall ensure that all instances of the TOE can be accounted for at all stages.
All instances of the TOE are transported and manipulated in a secure environment with accountability and traceability of all (good and bad) products.

Phase 7:
This End-User environment is defined in Smartcard PP 9806/v2.0 for the Atmel 3232CS part.

2.4 TOE logical phases

During its construction/usage, the TOE can be in one of 4 persistent logical phases:

- **Manufacturing state**, which represents the state of the Basic Software (BS) at chip manufacture prior to programming any data into the chip.
- **Pre-personalised state**, (User: Personaliser) which represents the state where the required data has already been programmed onto the chip, the required data includes: the MFOS operating system, the ISAM Application and the manufacturing data (e.g. manufacturing.objects such as transport keys). This state allows the first phase of personalisation data to be loaded using MFOS personalisation commands (prepersonalised.objects).
- **Personalised state** (User: Personalizer) allows the final phase of personalisation data to be loaded using MFOS personalisation commands (prepersonalised.objects).
- **Operational State** (User: POSTuser,HOPSuser (HOPSISMS and HOPSAMS)) represents the state at which the ISAM is ready for use in a Head Office Processor (HOPS) or a Point of Service Terminal (POST).

These phases are sorted above in logical order. The function to set personalisation state from one state to the next is under TOE control and is part of the MFOS operating system. In the operational state it is possible to perform a function to set pre-personalisation state, effectively deleting all data which has been loaded using the MFOS personalisation commands (This function can be configured to be available or not available in the Operational state)

2.5 TOE Intended Usage

In terms of the system context, the ISAM shall reside in every ITSO Point of Service Terminal (POST) and every ITSO Head Office Processor (HOPS).

The following diagram illustrates the logical ITSO system architecture model and the relationships between the various scheme participants.

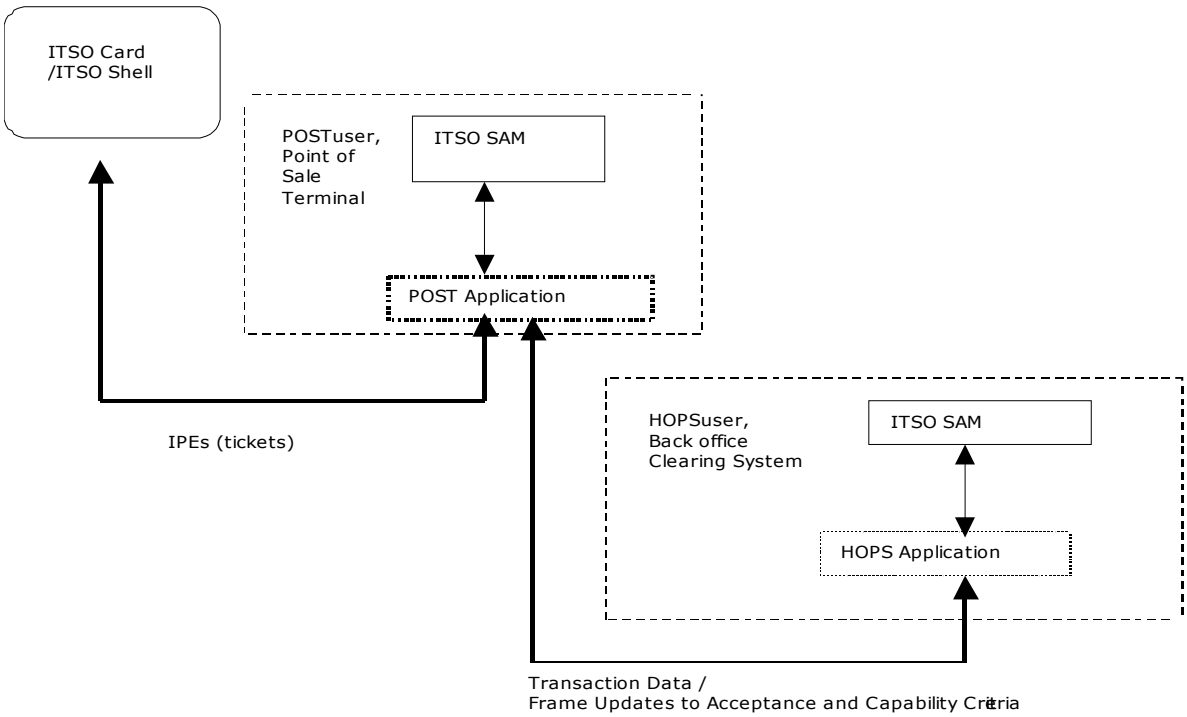

Figure 4

The ISAM acts as a secure:

- signature generation/verification engine,
- encryption /decryption engine,
- storage area for ITSO transaction data and transaction batch headers (this is an optional configuration of the ISAM),
- storage area and enforcer of ISAM specific ITSO scheme parameters. This includes Acceptance and Capability Criteria tables. These tables are used by the ISAM to only accept valid ITSO Product Entities (IPEs) and to enforce other rules regarding IPE processing. IPEs can be considered to be transport tickets.

The ITSO shell stores the IPEs and a Directory (DIR), the DIR contains a list of the available IPEs.

The POST and the ISAM manage IPE processing and storage/supply of transaction data.

The HOPS and the ISAM manage the final processing and verification of transaction data and also manage the Updates to an ISAM's Acceptance and Capability Criteria tables.

Furthermore there shall be two HOPSusers, HOPSISMS and HOPSAMS, HOPSAMS will be able to update all values in the ISAMs Acceptance and Capability Criteria tables, except Keys, HOPSISMS shall be able to update the Keys in the ISAMs Acceptance and Capability Criteria tables.

Transaction Data accumulated in the POST is signed by the ISAM for delivery to the HOPS, where the data is verified before processing. The HOPS generates delete parameters which allow the deletion of the transaction data from the ISAM that supplied them.

A configuration of an ISAM called PERSO (similar to that of a POST) shall be available to allow the personaliser of the ITSO Shell cards the required rights to seal a DIR for the first time
(Note the **PERSO** role refers to the personalizer of the SHELL card and is different to the **Personaliser** which refers to the personaliser of the ISAM),

2.5.1 TOE processes

During a transaction, a card with the ITSO Shell application communicates with the a POST which contains the ISAM, the functions that involve processing by the TOE can be split into five concatenated processes as follows:

1. Authenticating the card and opening the directory
2. IPE processing
3. Updating the directory and committing the transaction to the card
4. Ending the card session
5. Transaction record processing

These five main processes produce data flows between three entities, namely, an ITSO card, the Point of Service Terminal (POST) and the TOE (ISAM) and are illustrated below in Figure 5:

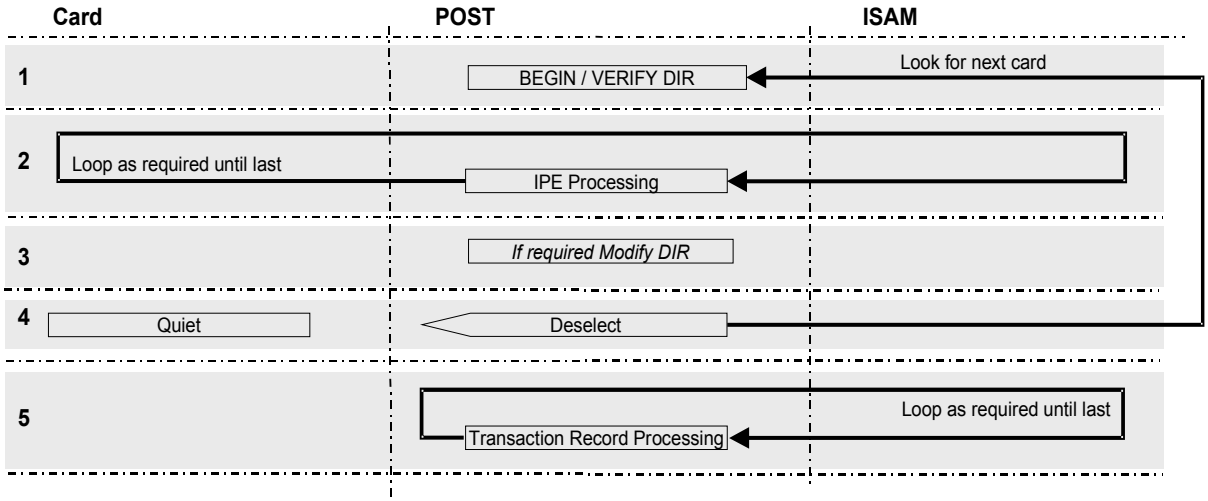

Figure 5

The details of TOE processing and commands, which may be named in the TOE Security Functional Requirements or Toe Security Functions of this document can be found in the 1.6Project ITSO, Interface Control Document, ITSO-ICD-001-L3E, Revision 6.2, 26th October 2004. Additionally any policies referred to by the TOE Security Functional Requirements shall be contained in the 1.6ISAM TOE Security Policy Model, ITSO-TSPM-001-L3E, Version 6.3, 13th April 2005.

Further details of the operation of the TOE can be found in the reference 1.6Project ITSO, Interface Control Document, ITSO-ICD-001-L3E, Revision 6.2, 26th October 2004.

2.6 General IT features of the TOE

The TOE IT Security functionalities consist of data storage and processing such as:

- arithmetical functions (e.g. incrementing/decrementing security counters, error counters or transaction sequence numbers),
- data communication (e.g. the receipt/transmission of ITSO data through the use of the WSAM and RSAM commands as detailed in ref 1.6Project ITSO, Interface Control Document, ITSO-ICD-001-L3E, Revision 6.2, 26th October 2004.)
- cryptographic operations (e.g. data encryption/decryption, digital signature generation/verification, message authentication code generation/verification, hashing, generation of random numbers).

3 TOE Security Environment

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.

3.1 Assets

Assets are security relevant elements of the TOE that include:

- the IC specifications, design, development tools and technology,
- the IC Dedicated software,
- the Smart Card Embedded Software including specifications, implementation and related documentation,
- the application data of the TOE (such as IC and system specific data, Initialisation data, IC pre-personalisation requirements and personalisation data,)

The TOE itself is therefore an asset.

Assets will be protected in terms of confidentiality, and integrity.

3.2 Assumptions

Assumptions described hereafter will be considered for a secure system implementation using Smart Card products containing the ITSO Shell and the TOE :

3.2.1 Assumptions on phase 1

A.DEV_ORG*	Procedures dealing with physical, personnel, organisational, technical measures for the confidentiality and integrity, of Smart Card Embedded Software (e.g. source code and any associated documents) and IC designer proprietary information (tools, software, documentation) shall exist and be applied in software development.

3.2.2 Assumptions on the TOE delivery process (phases 4 to 7)

Procedures shall guarantee the control of the TOE delivery and storage process and conformance to its objectives as described in the following assumptions:

A.DLV_PROTECT*	Procedures shall ensure protection of TOE material/information under delivery and storage.
A.DLV_AUDIT*	Procedures shall ensure that corrective actions are taken in case of improper operation in the delivery process and storage.

A.DLV_RESP* Procedures shall ensure that people dealing with the procedure for delivery have got the required skill.

.

3.2.3 Assumptions on phases 4 to 6

A.USE_TEST* It is assumed that appropriate functionality testing of the TOE is used in phases 4, 5 and 6.

A.USE_PROD* It is assumed that security procedures are used during all manufacturing and test operations through phases 4, 5, 6 to maintain confidentiality and integrity of the TOE and of its manufacturing and test data (to prevent any possible copy, modification, retention, theft or unauthorised use).

3.2.4 Assumption on phase 7

A.USE_DIAG* It is assumed that secure communication protocols and procedures are used between Smart Card and terminal.

3.3 Threats

The TOE as defined in Chapter 2 will counter the threats described hereafter; a threat agent wishes to abuse the assets either by functional attacks environmental manipulations, specific hardware manipulation, or through a combination of hardware and software manipulations or by any other types of attacks.

Threats are split in :

- - threats against which specific protection within the TOE is required (class I),
- - threats against which specific protection within the environment is required (class II).

3.3.1 Unauthorised full or partial cloning of the TOE

T.CLON* Functional cloning of the TOE (full or partial) is relevant to all phases of the TOE life-cycle, from phase 1 to phase 7, but only phases 1 and 4 to 7 are considered here, since functional cloning in phases 2 and 3 are purely in the scope of Smart Card IC PP (PP 9806, 1998). Generally, this threat is derived from specific threats combining unauthorised disclosure, modification or theft of assets at different phases.

3.3.2 Threats on phase 1

During phase 1, three types of threats have to be considered:

a) threats on the Smart Card Embedded Software and its development environment, such as unauthorised disclosure, modification or theft of the Smart Card Embedded Software and/or initialisation data at phase 1.

b) threats on the assets transmitted from the IC designer to the Smart Card software developer during the Smart Card ES development ;

c) threats on the Smart Card Embedded Software and initialisation data transmitted during the delivery process from the Smart Card software developer to the IC designer.

Unauthorised disclosure of assets
This type of threats covers unauthorised disclosure of assets by attackers who may possess a wide range of technical skills, resources and motivation. Such attackers may also have technical awareness of the product.

T.DIS_INFO* (type b)	Unauthorised disclosure of the assets delivered by the IC designer to the Smart Card Embedded Software developer, such as sensitive information on IC specification, design and technology, software and tools if applicable.
T.DIS_DEL* (type c)	Unauthorised disclosure of the Smart Card Embedded Software and any additional application data (such as IC pre-personalisation requirements) during the delivery to the IC designer.
T.DIS_ES1 (type a)	Unauthorised disclosure of ES (technical or detailed specifications, implementation code) and/or Application Data (such as secrets, or control parameters for protection system, specification and implementation for security mechanisms).
T.DIS_TEST_ES (type a and c)	Unauthorised disclosure of the Smart Card ES test programs or any related information.

Theft or unauthorised use of assets
Potential attackers may gain access to the TOE and perform operations for which they are not authorised. For example, an attacker may personalise, modify or influence the product in order to gain access to the Smart Card application system.

T.T_DEL* (type c)	Theft of the Smart Card Embedded Software and any additional application data (such as pre-personalisation requirements) during the delivery process to the IC designer.
T.T_TOOLS (type a and b)	Theft or unauthorised use of the Smart Card ES development tools (such as PC, development software, data bases).
T.T_SAMPLE2 (type a)	Theft or unauthorised use of TOE samples (e.g. bond-out chips with the Embedded Software).

Unauthorised modification of assets
The TOE may be subjected to different types of logical or physical attacks which may compromise security. Due to the intended usage of the TOE (the TOE environment may be hostile), the TOE security may be bypassed or compromised reducing the integrity of the TOE security mechanisms and disabling their ability to manage the TOE security. This type of threats includes the implementation of malicious Trojan horses.

T_MOD_DEL* (type c)	Unauthorised modification of the Smart Card Embedded Software and any additional application data (such as IC pre-personalisation requirements) during the delivery process to the IC designer.
T.MOD (type a)	Unauthorised modification of ES and/or Application Data or any related information (technical specifications).

3.3.3 Threats on delivery for/from phase 1 to phases 4 to 6

Threats on data transmitted during the delivery process from the Smart Card developer to the IC packaging manufacturer, the Finishing process manufacturer or the Personaliser.

T.DIS_DEL1	Unauthorised disclosure of Application Data during delivery to the IC Packaging manufacturer, the Finishing process manufacturer or the Personaliser.
T.DIS_DEL2	Unauthorised disclosure of Application Data delivered to the IC Packaging manufacturer, the Smartcard product manufacturer or the Personaliser.
T.MOD_DEL1	Unauthorised modification of Application Data during delivery to the IC Packaging manufacturer, the Smartcard product manufacturer or the Personaliser.
T.MOD_DEL2	Unauthorised modification of Application Data delivered to the IC Packaging manufacturer, the Smartcard product manufacturer or the Personaliser.

3.3.4 Threats on phases 4 to 7

During these phases, the assumed threats are described in three types :

- unauthorised disclosure of assets,
- theft or unauthorised use of assets,
- unauthorised modification of assets.

Unauthorised disclosure of assets
This type of threats covers unauthorised disclosure of assets by attackers who may possess a wide range of technical skills, resources and motivation. Such attackers may also have technical awareness of the product.

T.DIS_ES2 Unauthorised disclosure of ES and Application Data (such as data protection systems, memory partitioning, cryptographic programs and keys).

Theft or unauthorised use of assets
Potential attackers may gain access to the TOE and perform operations for which they are not allowed. For example, such attackers may personalise the product in an unauthorised manner, or try to fraudulently gain access to the Smart Card system.

T.T_ES Theft or unauthorised use of TOE.
(e.g. bound out chips with embedded software).

T.T_CMD Unauthorised use of instructions or commands
or sequence of commands sent to the TOE.

Unauthorised modification of assets
The TOE may be subjected to different types of logical or physical attacks that may compromise security. Due to the intended usage of the TOE (the TOE environment may be hostile), the TOE security parts may be bypassed or compromised reducing the integrity of the TOE security mechanisms and disabling their ability to manage the TOE security. This type of threat includes the implementation of malicious Trojan horses, Trapdoors, downloading of viruses or unauthorised programs.

T.MOD_LOAD Unauthorised loading of programs.

T.MOD_EXE Unauthorised execution of programs.

T.MOD_SHARE Unauthorised modification of program behavior by interaction of different programs.

T.MOD_SOFT* Unauthorised modification of Smart Card Embedded Software and Application Data.

The table below indicates the relationship between the phases of the Smart Card life cycle, the threats and the type of the threats:

Threats	Phase 1	Phase 4	Phase 5	Phase 6	Phase 7

T.CLON*	Class II	Class I	Class I	Class I	Class I
T.DIS_INFO*	Class II				
T.DIS_DEL*	Class II				
T.DIS_DEL1	Class II				
T.DIS_DEL2		Class II	Class II	Class II	
T.DIS_ES1	Class II				
T.DIS_TEST_ES	Class II				
T.DIS_ES2		Class I	Class I	Class I	Class I
T.T_DEL*	Class II				
T.T_TOOLS	Class II				
T.T_SAMPLE2	Class II				
T.T_ES		Class I	Class I	Class I	Class I
T.T_CMD		Class I	Class I	Class I	Class I
T.MOD_DEL*	Class II				
T.MOD_DEL1	Class II				
T.MOD_DEL2		Class II	Class II	Class II	
T.MOD	Class II				
T.MOD_SOFT*		Class I	Class I	Class I	Class I
T.MOD_LOAD		Class I	Class I	Class I	Class I
T.MOD_EXE		Class I	Class I	Class I	Class I
T.MOD_SHARE		Class I	Class I	Class I	Class I

3.4 Organisational Security policies

The TOE security objectives are derived purely from threats and assumptions therefore this section has been omitted.

4 Security objectives

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

- integrity and confidentiality of assets,
- protection of the TOE and associated documentation and environment during development and production phases.

4.1 Security Objectives for the TOE

The TOE shall use state of art technology to achieve the following IT security objectives, and for that purpose, when IC physical security features are used, the specification of those IC physical security features shall be respected. When IC physical security features are not used, the Security Objectives shall be achieved in other ways:

O.TAMPER_ES	The TOE must prevent tampering with its security critical parts. Security mechanisms have especially to prevent the unauthorised change of functional parameters, security attributes and secrets such as the life cycle sequence flags and cryptographic keys. The ES must be designed to avoid interpretations of electrical signals from the hardware part of the TOE.
O.CLON*	The TOE functionality must be protected from cloning.
O.OPERATE*	The TOE must ensure continued correct operation of its security functions.
O.FLAW*	The TOE must not contain flaws in design, implementation or operation.
O.DIS_MECHANISM2	The TOE shall ensure that the ES security mechanisms are protected against unauthorised disclosure.
O.DIS_MEMORY*	The TOE shall ensure that sensitive information stored in memories is protected against unauthorised disclosure.
O.MOD_MEMORY*	The TOE shall ensure that sensitive information stored in memories is protected against any corruption or unauthorised modification.

4.2 Security objectives for the environment

4.2.1 Objectives on phase 1

O.DEV_TOOLS*	The Smart Card ES shall be designed in a secure manner, by using exclusively software development tools (compilers assemblers, linkers, simulators, etc.) and software-hardware integration testing tools (emulators) that will result in the integrity of program and data.
O.DEV_DIS_ES	The Embedded Software developer shall use established procedures to control storage and usage of the classified development tools and documentation, suitable to maintain the integrity and the confidentiality of the assets of the TOE. It must be ensured that tools are only delivered and accessible to the parties authorised personnel. It must be ensured that confidential information on defined assets are only delivered to the parties authorised personnel on a need to know basis.
O.SOFT_DLV*	The Smart Card embedded software must be delivered from the Smart Card embedded software developer (Phase 1) to the IC designer through a trusted delivery and verification procedure that shall be able to maintain the integrity of the software and its confidentiality, if applicable.
O.INIT_ACS	Initialisation Data shall be accessible only by authorised personnel (physical, personnel, organisational, technical procedures).
O.SAMPLE_ACS	Samples used to run tests shall be accessible only by authorised personnel.

4.2.2 Objectives on the TOE delivery process (phases 4 to 7)

O.DLV_PROTECT* Procedures shall ensure the protection of TOE material/information, under delivery, including the following objectives :

- non-disclosure of any security relevant information,
- identification of the element under delivery,
- meet confidentiality rules (confidentiality level, transmittal form, reception acknowledgment),
- physical protection to prevent external damage,
- secure storage and handling procedures (including rejected TOE's),
- traceability of TOE during delivery including the following parameters:
  - origin and shipment details
  - reception, reception acknowledgement,
  - location material/information.

O.DLV_AUDIT* Procedures shall ensure that corrective actions are taken in case of improper operation in the delivery process (including any non conformance to the confidentiality convention) and highlight all non-conformance to this process.

O.DLV_RESP* Procedures shall ensure that people (shipping department, carrier, reception department) dealing with the procedure for TOE delivery have got the required skill, training and knowledge to meet the procedure requirements and be able to act fully in accordance with the above expectations.

4.2.3 Objectives on delivery from phase 1 to phases 4, 5 and 6

O.DLV_DATA	The Application Data will be delivered from the Smart Card embedded software developer (phase 1) either to the IC Packaging manufacturer, the Smartcard Product manufacturer or the Personaliser through a trusted delivery and verification procedure that shall be able to maintain the integrity and confidentiality of the Application Data.

4.2.4 Objectives on phases 4 to 6

O.TEST_OPERATE*	Appropriate functionality testing of the TOE shall be used in phases 4 to 6. During all manufacturing and test operations, security procedures shall be used through phases 4, 5 and 6 to maintain the confidentiality and integrity of the TOE and its manufacturing and test data.

4.2.5 Objectives on phase 7

O.USE_DIAG*	Secure communication protocols and procedures shall be used between the Smart Card and the terminal.

5 IT Security Requirements

5.1 Security requirements for the IT environment

The TOE has no asserted security dependencies on its IT environment. Therefore, there is no further statement or assumption made about the ST IT environment in this ST.

5.2 TOE Security Functional requirements

This chapter defines the functional requirements for the TOE using only functional requirements components drawn from the CC part 2.

The minimum strength level for the TOE security functions is " SOF-high" (Strength of Functions High).

5.2.1 Security audit analysis (FAU_SAA)

5.2.1.1 FAU_SAA.1 Potential Violation Analysis

FAU_SAA.1.1 The TSF shall be able to apply a set of rules in monitoring the audited events and based upon these rules indicate a potential violation of the TSP.

FAU_SAA.1.2 The TSF shall enforce the following rules for monitoring audited events :

a) Accumulation or combination of the following auditable events known to indicate a potential security violation :

1 Self-Test Failure,
2 Low Frequency of clock input,
3 High frequency of clock input,
4 Low voltage power supply,
5 High voltage power supply,
6 Low temperature,
7 High temperature,

b) Any other rules: *none.*

5.2.2 Cryptographic key management (FCS_CKM)

5.2.2.1 FCS_CKM.3 Cryptographic key access

FCS_CKM.3.1 / DES The TSF shall perform [**DES**] in accordance with a specified cryptographic key access method [**DESLoadKey, DES3LoadKey**] that meets the following: **[FIPS 46-3 1.6 , FIPS 811.6**].

FCS_CKM.3.1 / RSA The TSF shall perform [**RSA**] in accordance with a specified cryptographic key access method [**modExp, modExpCRT,** g**enCRTkeyset, genPublicModulus**] that meets the following: [**PKCS #1 v2.0 1.6**].

FCS_CKM.3.1 / MAC The TSF shall perform [**MAC**] in accordance with a specified cryptographic key access method [**DES3Sig**] that meets the following: [**ISO 9797-1 1.6**].

5.2.2.2 FCS_CKM.4 Cryptographic key destruction

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method, [**that zeroizes previously stored keying material and permanently prevents destroyed keys from being recovered**] that meets the following standards:

none

5.2.3 Cryptographic operations (FCS_COP)

5.2.3.1 FCS_COP.1 Cryptographic operations

FCS_COP.1.1 / DES The TSF shall perform [**encryption, decryption, signature, verification of signature]** in accordance with a specified cryptographic algorithm [**DES**] and cryptographic key sizes of [**56bits (DES), and 112 bits (triple-DES)**] that meet the following standards:
1. FIPS PUB 46-3 1.6
2. FIPS PUB 81 1.6

FCS_COP.1.1 / RSA The TSF shall perform [**encryption, decryption, signature, verification of signature]** in accordance with a specified cryptographic algorithm [**RSA]** and cryptographic key sizes [**of 1024, 1536]** bits that meet the following **standards:**
PKCS#1 v2.0, RSA Encryption Standard, RSA Laboratories, 1998, 1.6

FCS_COP.1.1 / SHA The TSF shall perform [**hashing]** in accordance with a specified cryptographic algorithm [**SHA-1]** and cryptographic key size [**No Key]** that meets the following standard:
1. FIPS PUB 180-1 1.6.

FCS_COP.1.1 / RNG The TSF shall perform [**Random Number Generation]** in accordance with a specified cryptographic algorithm [**No Algorithm]** and cryptographic key size [**No Key]** that meets the following standard:
1. None

FCS_COP.1.1 / MAC The TSF shall perform [**signature]** in accordance with a specified cryptographic algorithm [**MAC**] and cryptographic key size [**112 bits (triple-DES)]** that meet the following standard:
1. ISO 9797-1 1.6

5.2.4 Access Control Policy (FDP_ACC)

5.2.4.1 FDP_ACC.2 Complete Access control

FDP_ACC.2.1 /PP The TSF shall enforce the [**Pre-operational state access control policy**] on [**MFOS Personalisation System Commands and for all manufacturing.objects, prepersonalised.objects**], and all operations among subjects and objects covered by the SFP.

FDP_ACC.2.2 /PP The TSF shall ensure that all operations between any subject in the TSC and any object within the TSC are covered by an access control SFP.

FDP_ACC.2.1 / MFOS The TSF shall enforce the [**Mfos File System access control policy**] on [**ISAM Assets : as described in the TOE Security Policy Model ref: 1.6ISAM TOE Security Policy Model, ITSO-TSPM-001-L3E, Version 6.3, 13th April 2005.**], and all operations among subjects and objects covered by the SFP.

FDP_ACC.2.2/MFOS The TSF shall ensure that all operations between any subject in the TSC and any object within the TSC are covered by an access control SFP.

5.2.5 Access Control Functions (FDP_ACF)

5.2.5.1 FDP_ACF.1 Security attribute based access control

FDP_ACF.1.1/ PP The TSF shall enforce the [**Pre-operational state access control policy**] to objects based on [**lifecycle state byte**].

FDP_ACF.1.2 / PP The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [**If an interface command has an predefined appstate parameter which does not allow that command to be processed in the current lifecyle state (as indicated by the lifecycle state byte), then the command shall be rejected**].

FDP_ACF.1.3 / PP The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [**None**].

FDP_ACF.1.4 / PP The TSF shall explicitly deny access of subjects to objects based on the **[event where the hardware detector flags are set**].

FDP_ACF.1.1/ APP The TSF shall enforce the [**Pre-operational state access control policy**] to objects based on [**appstate parameter**].

FDP_ACF.1.2 / APP The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [**If an interface command has an predefined appstate parameter which does not allow that command to be processed in the current lifecyle state (as indicated by the lifecycle state byte) , then the command shall be rejected**].

FDP_ACF.1.3 / APP The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [**None**].

FDP_ACF.1.4 / APP The TSF shall explicitly deny access of subjects to objects based on the **[event where the hardware detector flags are set**].

FDP_ACF.1.1/ MFOS The TSF shall enforce the [**Mfos file system access control**] to objects based on [**the value of DF or EF AM and SC bytes**].

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

.

.

FDP_ACF.1.3 / MFOS The TSF shall explicitly authorise access of subjects to objects based on the following additional rules : [**None**].

FDP_ACF.1.4 / MFOS The TSF shall explicitly deny access of subjects to objects based on the **[event where the hardware detector flags are set**].

5.2.6 Data Authentication (FDP_DAU)

5.2.6.1 FDP_DAU.1 Basic Data Authentication

FDP_DAU.1.1 The TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of [**Batch Header(s), Transaction Record(s)].**

FDP_DAU.1.2 The TSF shall provide [**Verify MAC (VMAC) Batch Header, VMAC Transaction Record**] with the ability to verify evidence of the validity of the indicated information.

5.2.7 Export to outside TSF control (FDP_ETC)

5.2.7.1 FDP_ETC.1 Export of User Data without Security Attributes

FDP_ETC.1.1 / PP The TSF shall enforce the [**Pre-operational state access control policy**] when exporting user data, controlled under the SFP(s), outside of the TSC.

FDP_ETC.1.2 / PP The TSF shall export the user data without the user data's associated security attributes.

FDP_ETC.1.1 / MFOS	The TSF shall enforce the [**Mfos File System access control policy**] when exporting user data, controlled under the SFP(s), outside of the TSC.
FDP_ETC.1.2 / MFOS	The TSF shall export the user data without the user data's associated security attributes.

5.2.8 Import from Outside TSF Control (FDP_ITC)

5.2.8.1 FDP_ITC.1 Import of User Data without Security Attributes

FDP_ITC.1.1 / PP	**The TSF shall enforce the [Pre-operational state access control policy]** when importing user data, controlled under the SFP, from outside of the TSC**.**
FDP_ITC.1.2 / PP	The TSF shall ignore any security attributes associated with the user data when imported from outside the TSC.
FDP_ITC.1.3 / PP	The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TSC: **[none].**
FDP_ITC.1.1 / MFOS	The TSF shall enforce the [**Mfos File System access control policy**] when importing user data, controlled under the SFP, from outside of the TSC.
FDP_ITC.1.2 / MFOS	The TSF shall ignore any security attributes associated with the user data when imported from outside the TSC.
FDP_ITC.1.3 / MFOS	The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TSC: [**none**].

5.2.9 Residual Information protection (FDP_RIP)

5.2.9.1 FDP_RIP.1 Subset residual information protection

FDP_RIP.1.1/ ACCT	The TSF shall ensure that any previous information content of a resource is made unavailable upon the [**de-allocation of the resource to**] the following objects: **Acceptance and Capability Criteria Tables and Key Tables**
FDP_RIP.1.1/TRSNT_OBJ	The TSF shall ensure that any previous information content of a resource is made unavailable upon the [**de-allocation of the resource to**] the following objects: **ITSOBuffer temporary objects**

5.2.10 Stored data integrity (FDP_SDI)

5.2.10.1 FDP_SDI.2 Stored data integrity monitoring and action

FDP_SDI.2.1/EEPROM	The TSF shall monitor user data stored within the TSC for [**EEPROM integrity errors**] on all objects, based on the following attributes:[**EEPROMCRCFlag**].
FDP_SDI.2.2/EEPROM	Upon detection of a data integrity error, the TSF shall **report an error**.
FDP_SDI.2.1/FLASHROM	The TSF shall monitor user data stored within the TSC for [**Flash ROM integrity errors**] on all objects, based on the following attributes:[**FLASHCRCFlag**].
FDP_SDI.2.2/FLASHROM	Upon detection of a data integrity error, the TSF shall [**report an error**].

5.2.11 Authentication failures (FIA_AFL)

5.2.11.1 FIA_AFL.1 Basic authentication failure handling

FIA_AFL.1.1 / VERIFY_ITSO	The TSF shall detect when [**Personaliser or HOPSISMS defined number of]** unsuccessful certification attempts occur related to [**Verify_ITSO**].
FIA_AFL.1.2 / VERIFY_ITSO	When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF shall [**return an error**].
FIA_AFL.1.1 / FRAME	The TSF shall detect when [**Personaliser or HOPSISMS defined number of]** unsuccessful certification attempts occur related to [**UPDATE_FRAME**].
FIA_AFL.1.2 / FRAME	When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF shall [**block UPDATE_FRAME**]
FIA_AFL.1.1 / PERSOMAC	The TSF shall detect when [**Personaliser defined number of**] unsuccessful authentication attempts occur related to [setting of the personalization state].
FIA_AFL.1.2 / PERSOMAC	When the defined number of unsuccessful authentications attempts has been met or surpassed, the TSF shall [**block** setting of the personalization state].
FIA_AFL.1.1 / PREPERSOMAC	The TSF shall detect when [**Personaliser defined number of**] unsuccessful authentication attempts occur related to [setting of the personalization state].
FIA_AFL.1.2 / PREPERSOMAC	When the defined number of unsuccessful authentications attempts has been met or surpassed, the TSF shall [**block** setting of the personalization state].

5.2.12 User attribute definition (FIA_ATD)

5.2.12.1 FIA_ATD.1 User attribute definition

FIA_ATD.1.1 The TSF shall maintain the following list of security attributes belonging to individual users: **ISAM_DATA_FILE_USER=POSTUSER or HOPSAMS or HOPSISMS or PERSO.**

5.2.13 User Authentication (FIA_UAU)

5.2.13.1 FIA_UAU.1 Timing of authentication

FIA_UAU.1.1 The TSF shall allow [**TSF mediated actions of the following list**] to be performed on behalf of the user before the user is authenticated.
WSAM, RSAM, VERIFY_ISAM_ID, BEGIN, OPEN_IPE, VERIFY_ITSO, EXTERNAL AUTHENTICATE, VERIFY_ITSO, MODIFY_IPE, MODIFY_VALUE_IPE, CREATE_IPE, DELETE_IPE, WDIR, END, IMAC, LBATCH, VTRANS_MAC, VBATCH_MAC, UPDATE FRAME, CREATE FRAME, READPK, SELFTEST, SEARCH_ITSO. (See 1.6Project ITSO, Interface Control Document, ITSO-ICD-001-L3E, Revision 6.2, 26th October 2004.)]

FIA_UAU.1.2 The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

5.2.13.2 FIA_UAU.3 Unforgeable authentication

FIA_UAU.3.1 The TSF shall [**detect and prevent]** use of authentication data that has been forged by any user of the TSF.

FIA_UAU.3.2 The TSF shall [**detect and prevent]** use of authentication data that has been copied from any other user of the TSF.

5.2.13.3 FIA_UAU.4 Single-use Authentication Mechanisms

FIA_UAU.4.1/BATCH_HEADER The TSF shall prevent reuse of authentication data related to [**deleting Batch Headers & Transaction Records**].

5.2.14 User identification (FIA_UID)

5.2.14.1 FIA_UID.1 Timing of identification

FIA_UID.1.1 The TSF shall allow [**TSF mediated actions of the following list**] to be performed on behalf of the user before the user is identified.
Verify_ISAM_ID

FIA_UID.1.2 The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

5.2.15 User-subject Binding (FIA_USB)

5.2.15.1 FIA_USB.1 User-subject binding

FIA_USB.1.1 The TSF shall associate the appropriate user security attributes with subjects acting on behalf of that user.

5.2.16 Management of function in the TSF (FMT_MOF)

5.2.16.1 FMT_MOF.1 Management of security functions behavior

FMT_MOF.1.1 The TSF shall restrict the ability to [**disable**] the functions: [**Personalisation Commands:**]
to [**Personaliser**].

5.2.17 Management of security attributes (FMT_MSA)

5.2.17.1 FMT_MSA.1 Management of security attributes

FMT_MSA.1.1 / MFOS	The TSF shall enforce the [**Mfos file system security policy]** to restrict the ability to [**change_default, query, modify or delete**] **t**he security attributes [**of MFOS File system files]** to [**Personaliser]**.
FMT_MSA.1.1 / PP	The TSF shall enforce the [**Pre-operational state access control policy]** to restrict the ability to [**change_default, query, modify or delete**] **t**he security attributes [**lifecycle state byte]** to [**Personaliser]**.
FMT_MSA.1.1 / APP	The TSF shall enforce the [**Pre-operational state access control policy]** to restrict the ability to [**change_default, query, modify or delete**] **t**he security attributes [**appstate]** to [**No Users]**.

5.2.17.2 FMT_MSA.2 Secure security attributes

FMT_MSA.2.1	The TSF shall ensure that only secure values are accepted for security attributes.

5.2.17.3 FMT_MSA.3 Static attribute initialisation

FMT_MSA.3.1/MFOS	The TSF shall enforce the [**Mfos file system security policy]** to provide [**restrictive**] default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2/MFOS	The TSF shall allow the [**Personaliser**] to specify alternative initial values to override the default values when an object or information is created.
FMT_MSA.3.1/PP	The TSF shall enforce the [**Pre-operational state access control policy]** to provide [**restrictive**] default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2/PP	The TSF shall allow the [**Personaliser**] to specify alternative initial values to override the default values when an object or information is created.
FMT_MSA.3.1/APP	The TSF shall enforce the [**Pre-operational state access control policy]** to provide [**restrictive**] default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2/APP	The TSF shall allow the [**No User**] to specify alternative initial values to override the default values when an object or information is created.

5.2.18 Management of TSF data (FMT_MTD)

5.2.18.1 FMT_MTD.1 Management of TSF data

FMT_MTD.1.1 / File	The TSF shall restrict the ability to [**modify, or clear**] the [**authentication error counters**] to the [**HOPSuser (HOPSISMS and HOPSAMS), POSTuser, PERSO, Personaliser**].
FMT_MTD.1.1 / LifeCycle	The TSF shall restrict the ability to [**modify, or clear**] the [**life cycle state byte]** to the [**Personaliser**]
FMT_MTD.1.1 / AppConfig	The TSF shall restrict the ability to [**modify, or clear**] the [**ISAM_DATA_FILE**] to the [**Personaliser or HOPSISMS**]
FMT_MTD.1.1 / Authenticationfailures	The TSF shall restrict the ability to [**modify**] the [**authentication error counters maximum limits**] to the [**Personaliser)**]

5.2.19 Security management roles (FMT_SMR)

5.2.19.1 FMT_SMR.1 Security roles

FMT_SMR.1.1	The TSF shall maintain the roles [**POSTuser, PERSO, HOPSuser(HOPSISMS and HOPSAMS), Personaliser**].
FMT_SMR.1.2	The TSF shall be able to associate users with roles.

5.2.20 Class FMT : Actions to be taken for management:

Function	Actions	Function	Actions	Function	Actions
FAU_SAA.1	NA	FIA_AFL.1	a)	FMT_MTD.1	a)
FCS_CKM.3	a)	FIA_ATD.1	a)	FMT_SMR.1	NA
FCS_CKM.4	a)	FIA_UAU.1	a)	FPR_UNO.1	NA
FCS_COP.1	NM	FIA_UAU.3	NM	FPT_FLS.1	NM
FDP_ACC.2	NM	FIA_UAU.4	NM	FPT_PHP.3	NA
FDP_ACF.1	a)	FIA_UID.1	a)	FPT_SEP.1	NM
FDP_DAU.1	a)	FIA_USB.1	a)	FPT_TDC.1	NM
FDP_ETC.1	NM	FMT_MOF.1	a)	FPT_TST.1	NA
FDP_ITC.1	a)	FMT_MSA.1	a)		
FDP_RIP.1	NA	FMT_MSA.2	NM		
FDP_SDI.2	NA	FMT_MSA.3	a)		

Management activity versus functional requirements,

legend:

the letter refers to the respective management defined in part 2 of CC V2.1

NM: No Management activity

NA: Not Applicable

5.2.21 Unobservability (FPR_UNO)

5.2.21.1 FPR_UNO.1 Unobservability

FPR_UNO.1.1/DES The TSF shall ensure that [**all users]** are unable to observe the operation [**DESENCRYPT/DESDECRYPT]** on [**DES] Data** by [**all users**]**.**

FPR_UNO.1.1/RSA The TSF shall ensure that [**all users]** are unable to observe the operation **modExp, modExpCRT, genCRTkeyset and genPublicModulus** on [**RSA] Data** by [**all users**]**.**

5.2.22 Fail secure (FPT_FLS)

5.2.22.1 FPT_FLS.1 Failure with preservation of secure state

FPT_FLS.1.1 The TSF shall preserve a secure state when the following types of failures occur :[**EEPROM integrity failure, Power loss while processing, chip reset while processing**]

5.2.23 TSF Physical protection (FPT_PHP)

5.2.23.1 FPT_PHP.3 Resistance to physical attack

FPT_PHP.3.1/Software The TSF shall resist [**Simple and Differential Power Analysis , Differential Fault Analysis attacks**] to the [**externally accessible interfaces of the smart card**] by responding automatically such that the TSP is not violated.

FPT_PHP.3.1/Hardware The TSF shall resist [**tampering of voltage, clock input frequency and temperature**] by responding automatically such that the TSP is not violated.

5.2.24 Domain separation (FPT_SEP)

5.2.24.1 FPT_SEP.1 TSF Domain separation

FPT_SEP.1.1 The TSF shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects.

FPT_SEP.1.2 The TSF shall enforce separation between the security domains of subjects in the TSC.

5.2.25 Inter-TSF basic data consistency (FPT_TDC)

5.2.25.1 FPT_TDC.1 Inter-TSF data consistency

FPT_TDC.1.1 The TSF shall provide the capability to consistently interpret [**Batch Headers, Transaction Records, Delete Parameters and Data frames]** when shared between the TSF and another trusted IT product.

FPT_TDC.1.2 The TSF shall use [**the definitions defined in reference 1.6 Project ITSO, Interface Control Document, ITSO-ICD-001-L3E, Revision 6.2, 26th October 2004.]** when interpreting the TSF data from another trusted IT product.

5.2.26 TSF self test (FPT_TST)

5.2.26.1 TSF Testing (FPT_TST.1)

FPT_TST.1.1 / VERIFYISAMID	The TSF shall run a suite of self tests [**at the conditions** [**Verify ISAM ID**]] to demonstrate the correct operation of the TSF.
FPT_TST.1.1 /REQUEST	The TSF shall run a suite of self tests [**after receiving a request by a HOPSuser(HOPSAMS, HOPSISMS), PERSO or a POSTuser],** at the conditions [**the ISAM_DATA_FILE attribute of the requesting user is set to HOPSuser (HOPSISMS and HOPSAMS), PERSO *or POSTuser***] to demonstrate the correct operation of the TSF.
FPT_TST.1.2	The TSF shall provide **[HOPSuser(HOPSISMS and HOPSAMS), PERSO *or POSTuser*]** with the capability to verify the integrity of TSF data.
FPT_TST.1.3	The TSF shall provide **[HOPSuser(HOPSISMS and HOPSAMS), PERSO *or POSTuser*]** with the capability to verify the integrity of the stored TSF executable code.

5.2.27 FPT_RVM.1 Non-bypassability of the TSP

5.2.27.1 Non-bypassability of the TSP (FPT_RVM.1)

FPT_RVM.1.1	The TSF shall ensure that TSP enforcement functions are invoked and succeed before each function within the TSC is allowed to proceed.

5.3 TOE Security Assurance Requirements

The ST is conformant to a pre-defined named assurance package as follows:

- EAL 4 augmented: the security assurance requirements are a proper superset of all assurance components in EAL 4.
- The augmentation relates to the requirement to meet the following components AVA_VLA.4 Highly resistant, ALC_DVS.2 Sufficiency of security measures and ADV_IMP.2 Implementation of the TSF.

6 TOE Summary Specification

6.1 TOE Security Functions

This section defines the TOE and Figure 6 specifies how they satisfy the TOE Security Functional Requirements.

A comprehensive list of the TOE Security Functions supplied by the hardware shall be defined in the 1.6 Atmel AT90SC3232CS Security Target Lite, Revision Date, 30th April 2004. which shall be compliant with 1.6 "Protection Profile, Smartcard Integrated Circuit", Version 2.0, Issue September 1998 PP/9806

6.1.1 Event Audit (SF1)

SF1 is a hardware Security Function. It shall provide event-logging functionality and allow monitoring of the following auditable events:

- The external clock signal goes outside acceptable bounds,
- Attempts to physically probe the device,
- Application program abnormal runaway occurs,
- The external voltage supply goes outside acceptable bounds,
- The ambient temperature goes outside acceptable bounds,
- Attempts to gain illegal access to reserved RAM memory locations,
- Attempts to gain illegal access to reserved EEPROM memory locations,
- Attempts to gain illegal access to reserved peripheral or I/O register locations,
- Attempts to execute illegal Read instruction to read the program memory from a non-supervisor program location,
- Attempts to move the RAM stack to an illegal RAM memory location,
- Attempts to execute an instruction opcode that is not implemented,
- Attempts to illegally write access the device's EEPROM,
- Attempts to gain illegal access to device supervisor modes;

6.1.2 SELFTEST Function (SF2)

SF2 tests the device memory (RAM, flash ROM (code) and EEPROM) and external Flash ROM (XMEM) integrity. Failures in the integrity are indicated by appropriate status word responses to the commands. Any failures detected by the SelfTest, Verify ISAM ID, testRAM, testROM, testXMEM and testEEPROM commands are also returned in subsequent ATRs.

6.1.3 DES Key Operation (SF3)

SF3 shall ensure that access to DES cryptographic primitives is in accordance with the standards defined in related documents (see references FIPS 46-3 1.6 , FIPS 811.6]. This operation is protected against inadvertent/malicious key disclosure.

6.1.4 RSA Operations (SF4)

SF4 shall ensure that access to RSA cryptographic primitives and keying material is in accordance with the standards defined in related documents (see reference PKCS#1 v2.0, RSA Encryption Standard, RSA Laboratories, 1998, 1.6). This operation is protected against inadvertent/malicious disclosure of private RSA key(s).

6.1.5 Cryptographic Key Destruction (SF5)

SF5 shall ensure that when keys stored in memory are deleted that the memory is set to the erased state such that no residual key material remains in memory. Furthermore the TOE shall also allow the replacement of keys by overwriting existing key material.

6.1.6 Generate SHA-1 Hash (SF6)

SF6 shall manage the secure generation of a message digest (hash), using the SHA-1 algorithm (as defined in reference 1.6FIPS PUB 180-1, Secure Hash Standard", April 17, 1995, National Institute of Standards and Technology), of the data string referenced in the input parameters.

6.1.7 Generate Random Number (SF7)

SF7 shall ensure the secure generation of a random data string. SF7 will employ a number of statistical checks to check output data for randomness. (A detailed description of these statistical tests is provided in reference FIPS PUB 140-1 1.6.

6.1.8 Lifecycle Access Control (SF8)

SF8 will manage the availability of the external interface commands for each of the lifecycle states.

6.1.9 MFOS File System Access Control (SF9)

SF9 will grant subjects permission to use the MFOS file system functions to access specific memory locations, based on the security attributes of the targeted file system objects (DFs and EFs) and the security conditions (SC) of the accessed function.

.

6.1.10 Create ITSO MAC (SF10)

SF10 shall generate a Message Authentication Code (MAC) that will be used to protect the validity/authenticity of Batch Headers and Transaction Records.

6.1.11 Delete File (SF11)

SF11 shall ensure that any previous information content of a File (DF or EF) is made permanently unavailable, following the file's erasure.

6.1.12 Clear ITSO Buffer (SF12)

SF 12 shall ensure that any previous information content of the ITSOBuffer of the TOE is made permanently unavailable, following the removal of any temporary object held in it.

6.1.13 RAM Security Counter (SF13)

SF13 shall decrement a RAM security counter whenever the execution of the Verify_ITSO command is unsuccessful, due to authentication failures. When the counter reaches zero, the use of the Verify_ITSO command will be prohibited. The counter will be returned to its initial state, following a cold reset.

6.1.14 EEPROM Security Counter (SF14)

SF14 shall decrement an EEPROM security counter whenever the execution of the [UPDATE_FRAME, SET PERSONALISATION STATE, SET PRE-PERSONALISATION STATE] commands is unsuccessful, because of authentication failures. When the counter reaches zero, the use of these commands will be prohibited.
Before the counter reaches zero and after a successful authentication, the EEPROM security counter will be reset to its maximum default value.

6.1.15 User Configuration (SF15)

SF15 shall limit the file commands/operations that can be performed by the TOE, based on the setting of its ISAM_DATA_FILE attribute (as HOPSAMS, or HOPSISMS, PERSO or POST).
The ISAM_DATA_FILE attribute will be set to HOPSAMS, HOPSISMS, PERSO or POST during personalisation. After the lifecycle state flag has been set to the "Operational" state the user configuration parameter can only be changed using an UPDATE FRAME.

6.1.16 Pre-Authentication Action (SF16)

SF16 shall:

- authenticate the origin of a new [data frame] sent to it before an UPDATE FRAME command is accepted and,

- authenticate the origin of [the delete parameters] sent to it, before the POLL command is accepted.

If the origin of these data is not configured as a [HOPSISMS or HOPSAMS] user, these commands will not be allowed to execute.
If the signature (MAC) is not verified, the commands will not be allowed to execute.

6.1.17 Initialisation Function (SF17)

SF17 The initialisation function shall ensure that the ISAM initialises in a secure manner after power on or reset and that when an EEPROM operation has been interrupted it either completes successfully or the affected EEPROM is restored to its original value.

6.1.18 Sequence Number (SF18)

DELETED.

6.1.19 Delete Parameter (SF19)

SF19 shall ensure that the Delete parameters (which form part of the command data of the POLL command) contain the unique ID for the Batch Header/Transaction Records that are to be deleted.
If the unique ID is not included in the Delete parameters, then the command will not be allowed to execute. An error message will be returned.

6.1.20 Verify_ISAM_ID (SF20)

At the start of a session, SF20 requires the user to supply the correct TOE-user password for that ISAM, before it allows the processing of any ITSO application interface commands.

6.1.21 Create File (SF21)

SF21 will only allow the setting of the security attributes associated with an object, during the creation of this object.
Object security attributes will not be changed/updated at any other stage in the lifecycle of an object.

6.2 Relationship between SFs and SFRs

The table below identifies the relationship between Security Functions (SFs) and Security Functional Requirements (SFRs), see Figure 6.
Note: SF18 is not present in the table below. This is not an error. The feature that was previously identified as SF18 is not required and has been deleted from the current version of the TOE.

		Security Functions																				
		SF1	SF2	SF3	SF4	SF5	SF6	SF7	SF8	SF9	SF10	SF11	SF12	SF13	SF14	SF15	SF16	SF17	SF19	SF20	SF21
Security Functional Requirements	FAU_SAA.1	X	X																		
	FCS_CKM.3			X	X						X										
	FCS_CKM.4	X				X						X					X				
	FCS_COP.1			X	X		X	X			X										
	FDP_ACC.2								X	X											
	FDP_ACF.1	X							X	X											
	FDP_DAU.1										X										
	FDP_ETC.1								X	X											
	FDP_ITC.1								X	X							X				
	FDP_RIP.1											X	X								
	FDP_SDI.2		X																		
	FIA_AFL.1													X	X						
	FIA_ATD.1															X					
	FIA_UAU.1								X								X				
	FIA_UAU.3										X		X				X		X		
	FIA_UAU.4																		X		
	FIA_UID.1								X											X	
	FIA_USB.1								X							X					
	FMT_MOF.1								X												
	FMT_MSA.1								X	X											
	FMT_MSA.2								X	X											X
	FMT_MSA.3								X	X											X
	FMT_MTD.1								X					X	X		X				
	FMT_SMR.1								X	X						X					
	FPR_UNO.1			X	X																
	FPT_FLS.1	X																X			
	FPT_PHP.3	X	X	X	X																
	FPT_SEP.1									X											
	FPT_TDC.1										X						X		X		
	FPT_TST.1		X																		
	FPT_RVM.1		X						X	X							X	X		X	

Figure 6

6.3 TOE Assurance Measures

This section defines the TOE Assurance measures and Table 6.2 specifies how they satisfy the TOE Security Assurance Requirements.

6.3.1 Configuration Management (SA1)

SA1 shall provide the document "SA1 - CONFIGURATION MANAGEMENT PROCESS REPORT" and its references.

6.3.2 Office and Computer Security (SA2)

SA2 shall provide the document " SA2 – OFFICE AND COMPUTER SECURITY PROCESS REPORT" and its references.

6.3.3 Packaging, Preservation and Delivery (SA3)

SA3 shall provide the document " SA3 – PACKAGING, PRESERVATION AND DELIVERY PROCESS" and its references.

6.3.4 Security Target (SA4)

SA4 shall provide the document " Ecebs ISAM/MFOS Security Target" and its references.

6.3.5 Life Cycle Model (SA5)

SA5 shall provide the document " SA5 – Life Cycle Model " and its references.

6.3.6 TOE Security Policy Model (SA6)

SA6 shall provide the document " TOE SECURITY POLICY MODEL (TSPM)" and its references.

6.3.7 Functional Specification (SA7)

SA7 shall provide the document "FUNCTIONAL SPECIFICATION " and its references.

6.3.8 High Level Design (SA8)

SA8 shall provide the document "HIGH LEVEL DESIGN (HLD)" and its references.

6.3.9 Low Level Design (SA9)

SA9 shall provide the document "LOW LEVEL DESIGN (LLD) " and its references.

6.3.10 Implementation (SA10)

SA10 shall provide the document "IMPLEMENTATION " and its references.

6.3.11 Traceability Analysis (SA11)

SA11 shall provide the document " TRACEABILITY ANALYSIS" and its references. This document (and its references) shall show the correspondence between:

- ST: TOE Security Functions and the Functional Specification: Functions.
- Functional Specification: Functions and the High Level Design: Subsystem
- High Level Design: Subsystems and the Low Level Design: Modules.
- Low Level Design: Modules and the Implementation: Code.

6.3.12 Development Tool Definition (SA12)

SA12 shall provide the document "DEVELOPMENT TOOL DEFINITION" and its references.

6.3.13 Deliverable Manuals (SA13)

SA13.1 shall provide the document " SA13.1 – Administrator guidance " and its references.
SA13.2 shall provide the document " SA13.2 – User guidance " and its references.
SA13.5 shall provide the document " SA13.3 – Installation, generation, and start-up procedures " and its references.

6.3.14 Validation of Analysis (SA14)

SA14 shall provide the document " VALIDATION OF ANALYSIS" and its references.

6.3.15 Functional Test (SA15)

SA15 shall provide the document " FUNCTIONAL TEST" and its references.

6.3.16 Test Coverage Analysis (SA16)

SA16 shall provide the document "TEST COVERAGE ANALYSIS " and its references.

6.3.17 Testing Depth Analysis (SA17)

SA17 shall provide the document "TESTING DEPTH ANALYSIS" and its references.

6.3.18 Evaluation Strength Analysis (SA18)

SA18 shall provide the document " Evaluation STRENGTH ANALYSIS" and its references.

6.3.19 Independent Test (SA19)

SA19 shall provide the document "INDEPENDENT TEST" and its references.

6.3.20 Security Resistance Analysis (SA20)

SA20 shall provide the document " SECURITY RESISTANCE ANALYSIS " and its references.

6.4 Relationship between Security Assurance Measures and Security Assurance Requirements

The table below identifies the relationship between Security Measures (SMs) and Security Assurance Requirements (SARs), see Figure 7

Security Assurance Requirements	SA1 Configuration Management	SA2 Office & Computer Security	SA3 Packaging, preservation & Delivery	SA4 Security Target	SA5 Lifecycle Model	SA6 TOE SPM	SA7 Functional Spec.	SA8 High Level Design	SA9 Low Level Design	SA10 Implementation	SA11 Traceability Analysis	SA12 Development tools	SA13 Deliverable Manuals	SA14 Validation of Analysis	SA15 Functional Test	SA16 Test Coverage	SA17 Testing Depth	SA18 Evaluation Strength	SA19 Independent Test	SA20 Security Resistance
ACM_AUT.1	X																			
ACM_CAP.4	X																			
ACM_SCP.2	X																			
ADO_DEL.2			X																	
ADV_FSP.2							X													
ADV_HLD.2								X												
ADV_IMP.2										X										
ADV_LLD.1									X											
ADV_RCR.1											X									
ADV_SPM.1						X														
AGD_ADM.1													X							
ADO_IGS.1													X							
AGD_USR.1													X							
ALC_DVS.2		X																		
ALC_LCD.1					X															
ALC_TAT.1												X								
ASE				X																
ATE_COV.2																X				
ATE_DPT.1																	X			
ATE_FUN.1															X					
ATE_IND.2																			X	
AVA_MSU.2														X						
AVA_SOF.1																		X		
AVA_VLA.4																				X

Figure 7

7 PP Claims

The ST fully conforms only to the requirements of 1.6"Protection Profile - Smart Card Integrated Circuit with Embedded Software", Version 2.0, Issue June 1999, registered at the French Certification Body under the number PP/9911

This ST extends the Protection Profile (PP) to include functional requirement FPT_RVM.1 as per the functional components identified 1.6 "Common Criteria for information Technology Security Evaluation, Part 2: Security Functional Requirements", August 1999, version 2.1, CCIMB-99-032.

8 Rationale

8.1 Introduction

This part of the ST presents the evidence to support the claim that this ST is a complete and cohesive set of requirements, that a conformant TOE would provide an effective set of IT security countermeasures within the security environment, and that the TOE summary specification addresses these requirements.

8.2 Security Objectives Rationale

This section demonstrates that the stated security objectives address all the security environment aspects identified. Each security objective is correlated to at least one threat or one assumption.

8.2.1 Threats and Security Objectives

The following tables show which security objectives counter which threats, phase by phase.

During phase 1, the Smart Card Embedded Software (ES) is developed and Application Data are specified for all other phases.

The TOE is a functional product designed during phase 1, considering that the only purpose of the Embedded Software is to control and protect the operation of the TOE during phases 4 to 7 (product usage). The global security requirements to consider in the TOE, during the development phase, are the security threats of the other phases.

T.CLON*	The TOE being constructed can be cloned, but also the construction tools and document can help clone it. During phase 1, since the product does not exist, it cannot contribute to countering the threat. For the remaining phases 4 to 7, the TOE participates in countering the threats.
T.DIS_INFO*	This threat addresses disclosure of sensitive information concerning security mechanisms implemented in the IC and/or in the ES and known by the software developer, in order to meet the overall security objectives of the TOE. Sensitive information are transmitted by the IC designer to the Smart Card Software developer during phase 1.

T.DIS_DEL*	This threat addresses disclosure of software or Application Data which is delivered, from phase 1 to phase 2 for software embedding. As the data is not yet implemented in the TOE, the threat can only be countered by environmental procedures.
T.DIS_DEL1	This threat addresses disclosure of software or data during delivery from phase 1 to phases 4 to 6. As the data is not yet implemented in the TOE, the threat can only be countered by environmental procedures.
T.DIS_DEL2	This threat addresses disclosure of software or data which is delivered from phase 1 to phases 4 to 6. As the data is not yet implemented in the TOE, the threat can only be countered by environmental procedures.
T.DIS_ES1	This threat addresses disclosure of ES and/or Application Data. Although the ES is created in phase 1, it is active throughout the life of the Smart Card, and therefore this threat can be carried out during any and all of phases 1 through 7. During phases 1 and 2, as the product does not yet exist, it cannot contribute to countering the threat.
T.DIS_TEST_ES	This threat addresses disclosure of the Smart Card ES test programs or any related information. Tests concerning the embedded software or software to be embedded are carried out in phase 1. This threat is countered by environmental procedures, of which the tests themselves are part.
T.T_DEL*	This threat addresses the theft of software or Application Data, which is delivered for software embedding, from phase 1 to phase 2. As the data is not yet implemented in the TOE, the threat can only be countered by environmental procedures.
T.T_TOOLS	This threat addresses the Theft or unauthorised use of the Smart Card ES development tools. TOE development tools are only used during phase 1, so this threat can exist only during phase 1. As the TOE does not yet exist, this threat is countered by environmental procedures.
T.T_SAMPLE2	This threat addresses the theft or unauthorised use of TOE samples. TOE samples are used only during phase 1, so this threat can exist only during phase 1. As the TOE does not yet exist, this threat is countered by environmental procedures.

T.MOD_DEL*	This threat addresses the modification of software or data which is delivered for software embedding, in phase 2.
T.MOD_DEL1	This threat addresses the modification of Application Data during delivery to the IC packaging manufacturer (phase 4), the Finishing process manufacturer (phase 5), and to the Personaliser (phase 6).
T.MOD_DEL2	This threat addresses the modification of Application Data which is delivered to the IC packaging manufacturer (phase 4), the Finishing process manufacturer (phase 5), and to the Personaliser (phase 6).
T.MOD	This threat addresses the modification of ES and/or Application Data. Modification of software and Application Data can be done during ES design in phase 1. Since the product does not yet exist, the threat can only be countered by environmental objectives.
T.MOD_SOFT*	This threat addresses the modification of ES and/or Application Data. Once developed, the ES and the Application Data can be modified during any of the phases 4 to 7.
T.DIS_ES2	This threat addresses the disclosure of ES and Application Data (such as data protection systems, memory partitioning, cryptographic programs and keys). which can compromise security. During phases 4 to 7, the TOE counters the unauthorised disclosure of the ES and Application Data.
T.T_ES	This threat addresses the unauthorised use of stolen cards during the different phases of the Smart Card life cycle as well as the misappropriation of rights of the Smart Cards.
T.T_CMD	This threat addresses the diversion of the hardware or the software, or both, in order to execute non authorised operations.
T.MOD_LOAD, T.MOD_EXE, T.MOD_SHARE	The loading, execution and modification of resident programs shall not endanger the security of the TOE. The TOE will prevent interference between applications.

8.2.2 Threats addressed by security objectives

8.2.2.1 Security objectives for the TOE

During phase 1, the TOE does not yet exist, there is no threat on the TOE itself.

For the phases 4 to 7, the following figure indicates that each threat is mapped to at least one security objective during the life of the TOE:

Threats/ Obj	TAMPER_ES	OPERATE*	FLAW*	DIS_ MECHANISM 2	DIS_ MEMORY*	MOD_ MEMORY*	CLON*
T.CLON*				**X**	**X**		**X**
T.DIS_ES2	**X**	**X**	**X**	**X**	**X**		
T.T_ES	**X**	**X**	**X**			**X**	
T.T_CMD	**X**	**X**	**X**			**X**	
T.MOD_SOFT*	**X**	**X**	**X**			**X**	
T.MOD_LOAD	**X**	**X**	**X**			**X**	**X**
T.MOD_EXE	**X**	**X**	**X**			**X**	**X**
T.MOD_SHARE	**X**	**X**	**X**			**X**	**X**

Figure 14

The TOE uses state of the art technology to achieve the following IT security objectives:

O.TAMPER_ES addresses the protection of the security critical parts of the TOE and protects them from any disclosure, either directly (by bypassing protections) or indirectly by interpretation of physical or logical behavior. This feature addresses the disclosure-centered threat **T.DIS_ES2**.

Security mechanisms prevent the unauthorised modification of security attributes and functional parameters, such as the life cycle state flags. This feature addresses the modification-oriented threats **T.MOD_SHARE** and **T.MOD_SOFT***.

The ES is designed to avoid interpretations of electrical signals from the hardware part of the TOE. These characteristics cover the currents, voltages, power consumption, radiation, or timing of signals during the processing activity of the TOE.
The TOE provides physical and logical security mechanisms to prevent fraudulent access to any sensitive data, such as passwords, cryptographic keys or authentication data. This covers illegal use or duplication of TOE: **T.T_ES**, **T.T_CMD, T.MOD_LOAD** and **T.MOD_EXE**.

O.CLON* addresses the threat of cloning the TOE, **T.CLON***.
This objective also limits the possibility to access any sensitive security relevant information of the TOE, and thus covers **T.MOD_LOAD, T.MOD_EXE** and **T.MOD_SHARE**.

O.OPERATE*	The TOE ensures the correct continuation of operation of its security functions. Security mechanisms prevent the fraudulent usage of an interruption or change in the sequence of the normal processing order with the aim of to avoid the TOE security protection measures. These interruptions or changes may be carried out either by physical or by logical actions (statically or dynamically). This objective covers the unauthorised change of security attributes managing the access to sensitive information which are expressed as **T.DIS_ES2, T.MOD_SHARE** and **T._MOD_SOFT***. It also counters the actions of skipping the internal protections of the TOE, which result in threats **T.T_ES**, **T.T_CMD, T.MOD_LOAD** and **T.MOD_EXE**.
O.FLAW*	Addresses the threats **T.DIS_ES2, T.T_ES, T.T_CMD, T.MOD_LOAD, T.MOD_EXE, T.MOD_SHARE** and **T._MOD_SOFT*** by preventing any unauthorised modification of the TOE which could lead to malfunctions in security mechanisms during its design, production or operation.
O.DIS_MECHANISM2	The TOE ensures that the security mechanisms are protected against unauthorised disclosure, to combat the threats **T.DIS_ES2** and **T.CLON***. The security mechanisms (both hardware and software) and their functionality are kept confidential.
O.DIS_MEMORY*	The TOE ensures that sensitive information stored in memories is protected against unauthorised access. Such disclosure manifest itself as threat **T.DIS_ES2**, and can lead to **T.CLON***. This applies both to secret and access controlled information.

O.MOD_MEMORY* The TOE ensures that sensitive information stored in Memories (ROM, RAM, EEPROM) is protected against any corruption or unauthorised modification, which covers threat **T.MOD_SOFT*** and modification by unauthorised loading which covers threats **T.MOD_LOAD**.

The TOE also ensures that any loss of integrity cannot endanger the security, especially in case of modification of system flags or security attributes, thus combating threats **T.MOD_EXE** and **T.MOD_SHARE**.

The TOE prevents the fraudulent modification of such information as lifecycle state flags to avoid reversing the card life cycle sequence to gain access to prohibited information. Such modifications are a first step to
realize threats **T.T_ES** or **T.T_CMD**.

8.2.2.2 Security objectives for the environment

The following figure maps the security objectives for the environment relative to the various threats, during phase 1:

Threats/ Obj	DEV_ TOOLS*	DEV_ DIS_ES	SOFT_ DLV*	INIT _ACS	SAMPLE_ ACS
T.CLON*	**X**	**X**	**X**	**X**	**X**
T.DIS_INFO*		**X**			
T.DIS_DEL*	**X**	**X**	**X**	**X**	
T.DIS_ES1	**X**	**X**		**X**	
T.DIS_TEST_ES	**X**	**X**	**X**		
T.T_DEL*			**X**		
T.T_TOOLS	**X**				
T.T_SAMPLE2					**X**
T.MOD_DEL*		**X**	**X**	**X**	
T.MOD		**X**		**X**	

Figure 15

O.DEV.TOOLS* The development tools provide for the integrity, availability and reliability of both programs and data. This specificity protects against cloning (threat **T.CLON***).

Information Technology equipment are used to develop, to test, debug, modify, load the ES and personalise the TOE. This equipment shall only be accessible only by authorised personnel. This covers threats based on illegal access to equipment or development information: **T.DIS_ES1, T.DIS_TEST_ES, T.T_TOOLS.**

O.DEV_DIS_ES The ES is designed in a secure manner and focuses on the integrity, availability and confidentiality of programs and data.

Confidential information (such as user manuals and general information on defined assets) is only delivered to the 3rd parties' authorised personnel. This covers the disclosure based threats: **T.DIS_INFO*, T.DIS_DEL*, T.DIS_ES1** and **T.DIS_TEST_ES**, and thus helps to combat **T.MOD, T.MOD_DEL*** and **T.CLON***.

O.SOFT_DLV* O.SOFT_DLV addresses all the threats applicable to the delivery of the Smart Card Embedded Software to the IC designer through the use of a trusted delivery and verification procedure (**T.T_DEL***) that maintains the integrity (**T.MOD_DEL* , T.MOD**) and the confidentiality of the ES, if applicable (**T.DIS_DEL***), and of initialisation data (**T.DIS_ES1**) and test information (**T.DIS_TEST_ES**). This contributes to combat the threat **T.CLON***.

O.INIT_ACS Initialisation Data are only delivered to the Personaliser's authorised personnel and measures are taken to ensure their integrity. This covers disclosure based threats **T.DIS_DEL*** and **T.DIS_ES1**.

It also covers the theft based threats **T.MOD_DEL*** and **T.MOD**. All of this contributes to combating **T.CLON***.

O.SAMPLE_ACS Samples used to run tests will only be accessible by authorised personnel in order to avoid illicit use of such samples. These samples are considered as sensitive parts, since they can be used (with the relevant loaded security parameters) as production-grade trusted TOEs. This covers threats **T.T_SAMPLE2** and **T.CLON***.

The following figure maps the security objectives for the environment relative to the various threats on delivery, during phases 4 to 6:

Threats/ Obj	DLV_DATA	TEST_OPERATE*
T.DIS_DEL1	**X**	
T.DIS_DEL2		**X**
T.MOD_DEL1	**X**	
T.MOD_DEL2		**X**

Figure 16

O.DLV_DATA Protects against disclosure or modification of Application Data, during the delivery to other manufacturers, and thus covers the threats **T.DIS_DEL1** and T**.MOD_DEL1**.

O.TEST_OPERATE Protects against disclosure or modification of Application Data delivered to other manufacturers and thus covers the threats **T.DIS_DEL2** and **T.MOD_DEL2**.

8.2.3 Assumptions and security objectives for the environment

This section demonstrates that the combination of the security objectives is suitable to satisfy the identified assumptions for the environment.

Each of the assumptions for the environment is addressed by security objectives. Figure 17 demonstrates which objectives contribute to the satisfaction of each assumption. For clarity, the table does not identify indirect dependencies. This section describes why the security objectives are suitable to satisfy each of the assumptions.

Phases		Phase 1			Delivery process for phases 4 to 7.			Phases 4 to 6	Phase 7
	Assumptions	DEV_DIS_ES	DEV_TOOLS*	SOFT_DLV*	DLV_PROTECT*	DLV_AUDIT*	DLV_RESP*	TEST_OPERATE*	USE_DIAG*
1	DEV_ORG*	**X**	**X**	**X**					
4 to 7	DLV_PROTECT*				**X**				
4 to 7	DLV_AUDIT*					**X**			
4 to 7	DLV_RESP*						**X**		
4 to 6	USE_TEST*							**X**	
4 to 6	USE_PROD*							**X**	
7	USE_DIAG*								**X**

Figure 17

8.3 Security Requirements Rationale

This section demonstrates that the set of security requirements (TOE and environment) is suitable to meet the security objectives.

8.3.1 Security functional requirements rationale

This section demonstrates that the combination of the security requirements objectives is suitable to satisfy the identified security objectives. Figure 18 demonstrates which security functional requirements contribute to the satisfaction of each TOE security objective. For clarity, the figure does not identify indirect dependencies.

Security Requirements	TAMPER_ES	OPERATE*	DIS_MECHANISM2	DIS_MEMORY*	MOD_MEMORY*	FLAW*	CLON*
EAL4 requirements						**X**	
FAU_SAA.1	**X**	**P**	**P**	**X**	**X**		
FCS_CKM.3	**X**	**P**		**P**	**P**		**P**
FCS_CKM.4	**X**	**P**		**P**	**P**		**X**
FCS_COP.1	**X**			**X**			**P**
FDP_ACC.2	**X**	**P**	**X**	**X**	**P**		**P**
FDP_ACF.1	**X**	**P**	**X**	**X**	**P**		**P**
FDP_DAU.1	**X**	**P**			**X**		**P**
FDP_ETC.1				**X**	**P**		
FDP_ITC.1				**X**			
FDP_RIP.1	**X**			**P**			
FDP_SDI.2		**P**			**X**		
FIA_AFL.1	**X**	**P**			**P**		**P**
FIA_ATD.1	**X**	**P**			**P**		
FIA_UAU.1	**X**			**X**	**X**		**P**
FIA_UAU.3	**X**			**X**	**X**		**P**
FIA_UAU.4	**X**			**X**	**X**		**P**
FIA_UID.1	**X**			**X**	**X**		**P**
FIA_USB.1	**X**			**X**	**X**		**P**
FMT_MOF.1	**X**	**X**	**X**	**P**	**P**		**P**
FMT_MSA.1	**X**	**P**	**X**	**P**	**P**		**P**
FMT_MSA.2	**X**	**P**	**X**	**P**	**P**		**P**
FMT_MSA.3	**X**	**P**	**X**	**P**	**P**		**P**
FMT_MTD.1				**X**	**X**		**P**
FMT_SMR.1	**X**	**X**					
FPR_UNO.1	**X**	**P**		**X**	**X**		**X**
FPT_FLS.1	**X**						
FPT_PHP.3	**X**	**X**	**X**	**X**	**X**		**X**
FPT_SEP.1	**X**		**X**	**X**			
FPT_TDC.1	**X**				**X**		
FPT_TST.1		**P**			**X**		
Additional requirements							
FPT_RVM.1	**X**	**P**					

Figure 18 – Mapping of SFRs and security objectives

Legend: P: partial; X: relevant.

This section describes why the security functional requirements are suitable to meet each of the TOE security objectives.

The assurance requirements contribute to the satisfaction of the O.FLAW* security objectives. They are suitable because they provide the assurance that the TOE is designed, implemented and operates so that the IT security requirements are correctly provided.

As the TOE is able to detect potential physical violation via sensors and related circuitry, and logical violation through TSF enforcing functions, **FAU_SAA.1** meets the security objectives O.TAMPER_ES, O.DIS_MEMORY*, O.MOD_MEMORY* and partially O.OPERATE* and O.DIS_MECHANISM2 as it monitors events to indicate a potential violation of the TSP.

Cryptographic support functional requirements **FCS_CKM.3** and **FCS_CKM.4** support controlled access to assets by means of key management and key destruction (in the case of illicit access or any attempt to steal sensitive information). These functions combine to meet the security objectives of O.TAMPER_ES, and participate in meeting O.OPERATE*, O.DIS_MEMORY*, O.MOD_MEMORY* and O.CLON* requirements.

FCS_COP.1 which supports data encryption or electronic signing controls access to the assets by means of authentication mechanisms and encryption. This function combines to meet the security objectives of O.TAMPER_ES, O.DIS_MEMORY* and
also contributes to O.CLON*.

Access control functional requirements , **FDP_ACC.2** and **FDP_ACF.1** control the conditions of access to assets and operations among subjects and objects . This fulfills the security objectives, O.TAMPER_ES, O.DIS_MECHANISM2, O.DIS_MEMORY* and partially O.OPERATE* and O.MOD_MEMORY*. They participate in the fulfillment of O.CLON*.

The Data authentication functional requirement **FDP.DAU.1** assures the objectives O.TAMPER_ES and O.MOD_MEMORY*. It contributes to the correct operation of TOE,
O.OPERATE, and O .CLON*.

The export to outside TSF control functional requirement **FDP_ETC.1**, contributes to the realisation of O.DIS_MEMORY*. It also contributes to the correct operation of the TOE, O.MOD_MEMORY*.

Sensitive information can be securely imported from outside in order to be processed or stored inside the TOE. The TSF control functional requirement **FDP_ITC.1**, contributes to the realisation of O.DIS_MEMORY*.

FDP_RIP.1 prevents access to residual sensitive information which was temporarily stored in memories (EEPROM, RAM) during previous states of processing. This functional requirement meets objectives O.TAMPER_ES and partially O.DIS_MEMORY*.

The **FDP_SDI.2** functional requirement meets O.MOD_MEMORY* objectives. It also
contributes to the correct operation of TOE which covers O.OPERATE* .

Identification and authentication functional requirements **FIA_AFL.1** and **FIA_ATD.1**, which manage illicit authentication attempts and related security attributes meet O.TAMPER_ES and partially O.OPERATE* and O.MOD MEMORY*. **FIA_AFL1** also contributes to the correct operation of the TOE, O.CLON*.

Identification and authentication functional requirements **FIA_UAU.1**, **FIA_UAU.3**,
FIA_UAU.4, **FIA_UID.1** and **FIA_USB.1** prevent unauthorised access to stored memory, and thus contribute to security objectives O.TAMPER_ES, O.DIS_MEMORY* and O.MOD_MEMORY*. They also partially contribute to the correct operation of the TOE, i.e. O.CLON*.

FMT_MOF.1 restricts the ability to modify the access conditions or the user rights This functional requirement meets O.TAMPER_ES, O.OPERATE*, O.DIS_MECHANISM2 and partially O.DIS_MEMORY*, O.MOD_MEMORY*, O.CLON* objectives.

Management of TSF data functional requirements **FMT_MSA.1**, **FMT_MSA.2** and **FMT_MSA.3** which control the usage, modification and deletion of the security attributes meet the O.TAMPER_ES, and O.DIS_MECHANISM2 objective and contribute to the correct operation of the TOE, O.OPERATE*, O.DIS_MEMORY*, O.MOD_MEMORY and O.CLON*.

The **FMT_MTD.1** security functional requirement controls the authorisation to access or modify sensitive information. This requirement meets O.DIS_MEMORY* and O.MOD_MEMORY* objectives and partially O.CLON*.

The **FMT_SMR.1** functional requirement meets O.TAMPER_ES and O.OPERATE* objectives.

The **FPR_UNO.1** functional requirement meets O.TAMPER_ES, O.DIS_MEMORY*, O.MOD_MEMORY*, and O.CLON* especially protecting against the observation of internal processes of the TOE. It provides protection against unauthorised disclosure of sensitive information during operation of the TOE, under control of the Embedded Software. Thus, it also contributes to O.OPERATE*.

The **FPT_FLS.1** functional requirement meets O.TAMPER_ES objectives.

The **FPT_PHP.3** (Resistance to physical attack) functional requirement meets the objectives O.TAMPER_ES, O.DIS_MEMORY*, O.MOD_MEMORY* and O.CLON*. FPT_PHP.3 also meets the requirements O.OPERATE* and O.DIS_MECHANISM2.
The **FPT_SEP.1** functional requirement meets O.TAMPER_ES, O.DIS_MECHANISM2 and O.DIS_MEMORY* objectives.

The **FPT_TDC.1** functional requirement meets O.MOD_MEMORY* and O.TAMPER_ES
objectives. The TOE shall interpret consistently the information coming from trusted IT products.

The **FPT_TST.1** functional requirement meets O.MOD_MEMORY* and partially O.OPERATE*. The suite of self tests allows the verification of the integrity of executable code and/or sensitive memory content.

The **FPT_RVM.1** functional requirement meets O.TAMPER_ES and contributes to O.OPERATE*. The TOE authentication functions will be successfully completed before any other functionality becomes available.

8.3.2 Security functional requirement dependencies

This section demonstrates that all dependencies between components of security functional requirements included in this Security Target are satisfied.

The assurance requirements specified in this ST are precisely as defined in EAL4 with several higher hierarchical components (ADV_IMP.2, ALC_DVS.2 and AVA_VLA.4). This is a known set of assurance components for which all dependencies are satisfied.

Figure 19 lists the Security Functional Requirements defined in this Security Target (including security requirements for the IT environment), their dependencies and whether they are satisfied by other security requirements defined in this Security Target.

Number	Security Functional Requirements	Dependencies	Line No
1	FAU_SAA.1 : Potential Violation Analysis	none	a
2	FCS_CKM.3 : Cryptographic Key Access	FDP_ITC.1, FCS_CKM.4, FMT_MSA.2	9, 3, 21
3	FCS_CKM.4 : Cryptographic Key Destruction	FDP_ITC.1 , FMT_MSA.2	9, 21
4	FCS_COP.1 : Cryptographic Operation	FDP_ITC.1, FCS_CKM.4, FMT_MSA.2	9, 3, 21
5	FDP_ACC.2 : Complete Access Control	FDP_ACF.1, FPT_RVM.1	6,31
6	FDP_ACF.1 : security attributes based Access Control Functions	FDP_ACC.1, FMT_MSA.3, FPT_RVM.1	H(5)[b], 22,31
7	FDP_DAU.1 : basic Data Authentication	none	
8	FDP_ETC.1 : Export of user data without security attributes	none	
9	FDP_ITC.1 : Import of user data without security attributes	FMT_MSA.3	22
10	FDP_RIP.1 : subset residual information protection	none	
11	FDP_SDI.2 : stored data integrity monitoring and action	none	
12	FIA_AFL.1 : basic authentication failure handling	FIA_UAU.1, FMT_MTD.1	14, 23
13	FIA_ATD.1 : user attribute definition	None	
14	FIA_UAU.1 : timing of authentication	FIA_UID.1, FPT_RVM.1	17,31
15	FIA_UAU.3 : unforgeable authentication	none	
16	FIA_UAU.4 : Single-use authentication mechanisms	none	
17	FIA_UID.1 : timing of identification	FPT_RVM.1	31
18	FIA_USB.1 : user-subject binding	FIA_ATD.1	13
19	FMT_MOF.1 : management of security functions behavior	FMT_SMR.1, FPT_RVM.1	24,31
20	FMT_MSA.1 : management of security attributes	FMT_SMR.1, FPT_RVM.1	24,31
21	FMT_MSA.2 : safe security attributes	ADV_SPM.1, FMT_MSA.1, FMT_SMR.1	by EAL4 20, 24
22	FMT_MSA.3 : safe attributes initialisation	FMT_MSA.1, FMT_SMR.1	20, 24
23	FMT_MTD.1 : management of TSF data	FMT_SMR.1, FPT_RVM.1	24,31
24	FMT_SMR.1 : security roles	FIA_UID.1	17

25	FPR_UNO.1 : Unobservability	None	
26	FPT_FLS.1 : failure with preservation of secure state	ADV_SPM.1	by EAL4
27	FPT_PHP.3 : Resistance to physical attacks	none	
28	FPT_SEP.1 : TSF Domain separation	none	
29	FPT_TDC.1 : inter-TSF basic TSF data consistency	none	
30	FPT_TST.1 : TSF testing	none	a
31	FPT_RVM.1 : Non-bypassability of the TSP	FIA_UID.1, FIA_USB.1, FMT_MSA.2, FMT_MSA.3, FPT_SEP.1	17,18,21, 22,28

Figure 19

a : dependencies are not met for reasons given below
b: H(5) means that the dependency is satisfied by a higher hierarchical component

The dependency of FAU_SAA.1 with FAU_GEN.1 is not applicable to the TOE ; the FAU_GEN.1 component forces many security relevant events to be recorded (due to
dependencies with other functional security components) and this is not achievable in a Smart Card since many of these events result in card being in an insecure state where recording of the event itself could cause a security breach. It is then assumed that the function FAU_SAA.1 may still be used and the specific audited events will have to be defined in the ST independently with FAU_GEN.1.

The dependency of FPT_TST.1 with FPT_AMT.1 is not clearly relevant for a Smart Card ; FPT_TST.1 is self-consistent for the TOE (hardware and software) and does not require the FPT_AMT.1 function (Abstract Machine Testing). The TOE software is not tested inside the scope of FPT_TST.1. In its relations with external devices, typically the card reader, the TOE is always the slave. This is why FPT_TST.1 is self consistent, and FPT_AMT.1 is not applicable.

Figure 19 shows that the functional component dependencies are satisfied by other functional component of the ST.

8.3.3 Strength of Function (SOF) Level rationale

The TOE is intended to be used as a distributed, unsupervised, smartcard-based, digital signature generation/verification, encryption/decryption "engine" for a transport ticketing scheme. Therefore, it is assumed that attackers with high attack potential (resources, expertise and opportunities) will target the TOE. The claimed Strength of Function level is "high" to assure that even these attackers cannot successfully attack the TOE and that a successful attack is beyond practicality.

8.3.4 Security Assurance Requirements Rationale

The assurance requirements for this Security Target are summarised in Figure 20:

Requirements	Name	Type
EAL4	Methodically Designed, Tested and Reviewed	Assurance level
ADV_IMP.2	Implementation of the TSF	Higher hierarchical component
ALC_DVS.2	Development Security Measures	Higher hierarchical component
AVA_VLA.4	Highly resistant	Higher hierarchical component

Figure 20

8.3.4.1 Evaluation Assurance Level Rationale

An assurance level of EAL4 was chosen for this TOE, since it is intended to defend against sophisticated attacks. The assurance level was chosen to meet the assurance expectations of digital signature, encryption/decryption applications in a distributed transport-ticketing scheme.

8.3.4.2 Assurance Augmentations Rationale

Additional assurance requirements (beyond the assurance level) are required, because of the definition of the TOE:

8.3.4.2.1 ADV_IMP.2 Implementation of the TSF

This assurance component is a higher hierarchical component to EAL4. ADV_IMP.2 has dependencies with ADV_LLD.1 "Descriptive Low-Level design", ADV_RCR.1 "Informal correspondence demonstration", ALC_TAT.1 "Well defined development tools".

These components are included in EAL4, and so their dependencies are satisfied.

8.3.4.2.2 ALC_DVS.2 Sufficiency of security measures

Development security is concerned with physical, procedural, personnel and other technical procedures that may be used in the development environment to protect the TOE.

This assurance component is a higher hierarchical component to EAL4. Because of the definition of the TOE, the sufficiency of these procedures to protect the confidentiality and the integrity of the TOE is assessed.

ALC_DVS.2 has no dependencies.

8.3.4.2.3 AVA_VLA .4 Highly resistant

The TOE is required to be highly resistant to penetration attacks, initiated by attackers that are thoroughly familiar with the specific implementation of the TOE. and with a high level of technical sophistication.

This requirement stems from the fact that a Smart Card can be easily placed in a hostile environment manned by experts, such as electronic laboratories.

This assurance requirement is achieved by the AVA_VLA.4 component.

AVA_VLA.4 has dependencies with ADV_FSP.1 "Informal functional specification", ADV_HLD.2 "Security enforcing high-level design", ADV_LLD.1 " Descriptive low level

design", ADV_IMP.1 " Subset of the implementation of the TSF ", AGD_ADM.1 "Administrator Guidance" and AGD_USR.1 " User Guidance ".

All these dependencies are satisfied by EAL4.

8.3.4.3 Security requirements are mutually supportive and internally consistent

The purpose of this section is to confirm that the security requirements are mutually
supportive and internally consistent.

For the security assurance requirements, this claim is supported by the fact that:

- EAL4 is an established set of mutually supportive and internally consistent assurance requirements,
- The dependencies analysis for the additional assurance components in section 8.3.4.2 has shown that the assurance requirements are mutually supportive and internally consistent (all the dependencies have been satisfied).

For the security functional requirements, this claim is supported by the fact that:

-The dependencies analysis for the functional requirements, described in sections 8.3.1 and 8.3.2, demonstrates mutual support and internal consistency between the functional requirements.

-Inconsistency between functional and assurance requirements only arises if there are functional-assurance dependencies which are not met, a situation which has been shown not to arise in the section 8.3.2 above, "Security functional requirements dependencies".

Therefore, the dependencies' analysis described above, confirms mutual support and internal consistency between the security functional requirements.

8.4 TOE Summary Specification rationale

This section demonstrates that the set of TOE security functions and assurance measures is suitable to meet the TOE security requirements.

8.4.1 Security Functions Rationale

This section demonstrates that the combination of the TOE security functions work together to satisfy the TOE security functional requirements. Figure 21 demonstrates which security functions contribute to the satisfaction of each SFR.

		Security Functions																					
		SF1	SF2	SF3	SF4	SF5	SF6	SF7	SF8	SF9	SF10	SF11	SF12	SF13	SF14	SF15	SF16	SF17		SF20	SF21
Security Functional Requirements	FAU_SAA.1	X	P																		
	FCS_CKM.3			X	X						X										
	FCS_CKM.4	X				X						X					X				
	FCS_COP.1			X	X		X	X			X										
	FDP_ACC.2								X	X											
	FDP_ACF.1	X							X	X											
	FDP_DAU.1										X										
	FDP_ETC.1								X	X											
	FDP_ITC.1								X	X							X				
	FDP_RIP.1											X	X								
	FDP_SDI.2		X																		
	FIA_AFL.1													X	X						
	FIA_ATD.1															X					
	FIA_UAU.1								X								X				
	FIA_UAU.3										X		P				X		P		
	FIA_UAU.4																		X		
	FIA_UID.1								X											X	
	FIA_USB.1								P							X					
	FMT_MOF.1								X												
	FMT_MSA.1								X	X											
	FMT_MSA.2								X	X											X
	FMT_MSA.3								X	X											X
	FMT_MTD.1								X					X	X		X				
	FMT_SMR.1								P	X						X					
	FPR_UNO.1			X	X																
	FPT_FLS.1	P																X			
	FPT_PHP.3	X	P	X	X																
	FPT_SEP.1									X											
	FPT_TDC.1										X						X		X		
	FPT_TST.1		X																		
	FPT_RVM.1		X						X	X							X	X		X	

Figure 21

Legend: P: partial; X: relevant.

.:

Entry	Meaning

Figure 22

Figure 23

.

8.4.2 Strength of Function Claims Rationale

The claimed Strength of Function rating for the probabilistic/permutational security mechanisms described in this ST is “high”.

The TOE described in this Security Target is intended to operate in an unattended environment that may be under the control of a potential attacker. Further, the TOE may be exposed to this environment for considerable periods of time. Since the TOE will represent a high-profile smartcard transport system, it is likely to attract the attention of highly capable attackers who will have an opportunity to attract their target repetitively.

Any statistical or probabilistic mechanisms in the TOE may be subjected to prolonged analysis and attack in the normal course of their operation. Therefore, such mechanisms are claimed to be as resistant to attacks as possible, dictating a strength of function high rating for the Security Functions of this ST.

8.4.3 Security Assurance Measures Rationale

This section demonstrates that the stated security assurance measures are compliant with the assurance requirements. The table below identifies how each Security Measures complies with one (or more) Security Assurance Requirements:

Security Assurance Requirements	SA1 Configuration Management	SA2 Office & Computer Security	SA3 Packaging, preservation & Delivery	SA4 Security Target	SA5 Lifecycle Model	SA6 TOE SPM	SA7 Functional Spec.	SA8 High Level Design	SA9 Low Level Design	SA10 Implementation	SA11 Traceability Analysis	SA12 Development tools	SA13 Deliverable Manuals	SA14 Validation of Analysis	SA15 Functional Test	SA16 Test Coverage	SA17 Testing Depth	SA18 Evaluation Strength	SA19 Independent Test	SA20 Security Resistance
ACM_AUT.1	X																			
ACM_CAP.4	X																			
ACM_SCP.2	X																			
ADO_DEL.2			X																	
ADV_FSP.2							X													
ADV_HLD.2								X												
ADV_IMP.2									X											
ADV_LLD.1										X										
ADV_RCR.1											X									
ADV_SPM.1						X														
AGD_ADM.1												X								
ADO_IGS.1												X								
AGD_USR.1												X								
ALC_DVS.2		X																		
ALC_LCD.1					X															
ALC_TAT.1											X									
ASE				X																
ATE_COV.2															X					
ATE_DPT.1																X				
ATE_FUN.1														X						

ATE_IND.2																		X	
AVA_MSU.2													X						
AVA_SOF.1																	X		
AVA_VLA.4																			X

Figure 24

Configuration Management (SA1)
SA1 shall provide the document "SA1 - CONFIGURATION MANAGEMENT PROCESS REPORT" and its references.

Office and Computer Security (SA2)
SA2 shall provide the document " SA2 – OFFICE AND COMPUTER SECURITY PROCESS REPORT" and its references.

.

Packaging, Preservation and Delivery (SA3)
SA3 shall provide the document " SA3 – PACKAGING, PRESERVATION AND DELIVERY PROCESS" and its references.

Security Target (SA4)
SA4 shall provide the document " Ecebs ISAM/MFOS Security Target" and its references.

Life Cycle Model (SA5)
SA5 shall provide the document " SA5 – Life Cycle Model " and its references.

TOE Security Policy Model (SA6)
SA6 shall provide the document " TOE SECURITY POLICY MODEL (TSPM)" and its references.

Functional Specification (SA7)
SA7 shall provide the document "FUNCTIONAL SPECIFICATION " and its references.

High Level Design (SA8)
SA8 shall provide the document "HIGH LEVEL DESIGN (HLD)" and its references.

Low Level Design (SA9)
SA9 shall provide the document "LOW LEVEL DESIGN (LLD) " and its references.

.

Implementation (SA10)

SA10 shall provide the document "IMPLEMENTATION " and its references.

.

Traceability Analysis (SA11)

SA11 shall provide the document " TRACEABILITY ANALYSIS" and its references. This document (and its references) shall show the correspondence between:

.

Development Tool Definition (SA12)
SA12 shall provide the document "DEVELOPMENT TOOL DEFINITION" and its references.

Deliverable Manuals (SA13)
SA13.1 shall provide the document " SA13.1 – Administrator guidance " and its references.

Validation of Analysis (SA14)
SA14 shall provide the document " VALIDATION OF ANALYSIS" and its references.

.

Functional Test (SA15)
SA15 shall provide the document " FUNCTIONAL TEST" and its references.

Test Coverage Analysis (SA16)
SA16 shall provide the document "TEST COVERAGE ANALYSIS " and its references.

Testing Depth Analysis (SA17)
SA17 shall provide the document "TESTING DEPTH ANALYSIS" and its references.

Evaluation Strength Analysis (SA18)
SA18 shall provide the document " Evaluation STRENGTH ANALYSIS" and its references.

Independent Test (SA19)
SA19 shall provide the document "INDEPENDENT TEST" and its references.

Security Resistance Analysis (SA20)
SA20 shall provide the document " SECURITY RESISTANCE ANALYSIS " and its references.

8.5 PP Claims Rationale

This ST claims formal conformance with 1.6"Protection Profile - Smart Card Integrated Circuit with Embedded Software", Version 2.0, Issue June 1999, registered at the French Certification Body under the number PP/9911. The security objectives of this ST are identical to those of PP/9911.
The Assurance Requirements of this ST are identical to those stated in PP/9911.

An extra Security Functional Requirement (not included in PP/9911) has been added to this ST, i.e. FPT_RVM.1 (non-bypassability of the TOE Security Policy). This additional functional component is identified in 1.6"Common Criteria for information Technology Security Evaluation, Part 2: Security Functional Requirements", August 1999, version 2.1, CCIMB-99-032.

The addition of FPT_RVM.1, whose dependencies are all satisfied by the remaining Security Functional Requirements and which is satisfied by the stated Security Functions, is intended to ensure that the functions enforcing the TOE's Security Policies cannot be altogether bypassed during the life of the TOE. The TOE will be operating in an unsupervised, distributed environment. The occurrence of an attack that bypasses all the security measures introduced by the TOE developer will have an adverse effect on the credibility/survivability of the ITSO ticketing system that the TOE protects.

Therefore FPT_RVM.1 and the claimed "high" strength rating of the security functions that satisfy it, are an necessary addition to this ST, supplementing all the security objectives and requirements stated in 1.6"Protection Profile - Smart Card Integrated Circuit with Embedded Software", Version 2.0, Issue June 1999, registered at the French Certification Body under the number PP/9911.

9 Annex A - Glossary of Terms

9.1 Common Criteria Terminology

This section contains only those CC Terms, which are used in this ST. A larger Glossary section containing all the terms that are used in a specialised way in Common Criteria can be found in Section 2 of 1.6"Common Criteria for information Technology Security Evaluation, Part 1: Introduction and general model", August 1999, version 2.1, CCIMB-99-031. The majority of terms used in this ST are used either according to their accepted dictionary definitions or commonly accepted definitions found in ISO security glossaries or other well-known collections of security terms.

Assets	Information or resources to be protected by the countermeasures of the TOE.
Assurance	Ground for confidence that an entity meets its security objectives.
Attack potential	The perceived potential for success of an attack, should an attack be launched, expressed in terms of an attacker's expertise, resources and motivation.
Augmentation	The addition of one or more assurance component(s) from Common Criteria to an EAL or assurance package.
Authentication data	Information used to verify the claimed identity of a user.
Authorised user	A user who may, in accordance with the TSP, perform an operation.
Component	The smallest selectable set of elements that may be included in a Protection Profile, a Security Target or package.
Dependency	A relationship between requirements such that the requirement that is depended upon must normally be satisfied for the other requirements to be able to meet their objectives.
Evaluation Assurance	A package consisting of assurance components from Common Criteria (CC) that represents a point on the CC predefined assurance scale.
Identity	A representation (e.g. a string) uniquely identifying an authorised user.
Object	An entity within the TSC that contains or receives information and upon which subjects perform operations.
Organisational Security Policy	The set of security rules, procedures, practices, and guidelines imposed by ITSO upon its operations.
Package	A reusable set of either functional or assurance components (e.g. an EAL), combined together to satisfy a set of identified security objectives.
Protection Profile (PP)	An implementation-independent set of security requirements that meets specific consumer needs.
Role	A predefined set of rules establishing the allowed interactions between a user and the TOE.
Security attribute	Information associated with subjects, users and/or objects that is used for the enforcement of the TSP.
Security Function (SF)	A part or parts of the TOE that have to be relied upon for enforcing a closely related subset of the rules from the TSP.

Security Function Policy (SFP)	The security policy enforced by an SF.
Security objective	A statement of intent to counter identified threats and/or satisfy identified organisation security policies and assumptions.
Security Target (ST)	A set of security requirements and specifications to be used as the basis for evaluation of an identified TOE.
Strength of Function (SOF)	A qualification of a TOE security function expressing the minimum efforts assumed necessary to defeat its expected security behavior by directly attacking its underlying security mechanisms.
SOF-high	A level of the TOE strength of function where analysis shows that the function provides adequate protection against a deliberately planned or organised breach of TOE security by attackers possessing a high attack potential.
Subject	An entity within the TSC that causes operations to be performed.
Target of Evaluation (TOE)	An IT product or system, including its associated administrator and user guidance documentation that is the subject of an evaluation.
TOE Security	A set consisting of all hardware, software, and firmware of the TOE that Functions (TSF) must be relied upon for the correct enforcement of the TSP.
TOE Security Policy (TSP)	A set of rules that regulates how assets are managed, protected and distributed within a TOE.
TOE Security Policy Model	A structured representation of the security policy to be enforced by the TOE.
Transfers outside TSF control	Communication of data to entities not under control of the TSF.
TSF Scope of Control (TSC)	The set of interactions that can occur with or within a TOE and are subject to the rules of the TSP.
User	Any entity (human user, resident added application, or external IT entity) outside the TOE that interacts with the TOE.
User data	Data created by and for the user, which does not affect the operation of the TSF.

9.2 Smart Card Terminology

This section contains only those terms that are used in this ST in a way specific to the smart card industry.

Application	A file structure, directory entries and security schema loaded onto a smart card to perform a particular function, e.g. transaction generation/verification and audit "engine" for the ITSO transport scheme. There may be one or more applications on a card.
Application Data	IC and system specific data, Initialisation data, IC pre-personalisation requirements and personalisation data
APDU	Application Protocol Data Unit
Application Software (AS)	The part of ES in charge of the Application of the Smart Card IC.
Answer to Reset (ATR)	As defined in 1.6ISO/IEC 7816-3, Identification Cards-Integrated circuit(s) cards with contacts-Part 3: Electronic Signals and Transmission Protocols.
Basic Software (BS)	The part of ES in charge of the generic functions of the Smart Card IC such as Operating System, general routines and Interpretors.
Card embedder	A manufacturer who assembles a card and integrated circuit
Dedicated File (DF)	As defined in 1.6ISO/IEC 7816-3, Identification Cards-Integrated circuit(s) cards with contacts-Part 3: Electronic Signals and Transmission Protocols.
Differential Power Analysis (DPA)	A technique combining physical measurements of such things as power consumption with statistical signal processing techniques to identify IC operating details. DPA can, in some instances, provide information leading to recovery of internal operational parameters, keys, etc.
Electrically Erasable Programmable Read Only Memory (EEPROM)	A non-volatile memory technology where data can be electrically erased and rewritten.
Elementary File (EF)	As defined in 1.6ISO/IEC 7816-3, Identification Cards-Integrated circuit(s) cards with contacts-Part 3: Electronic Signals and Transmission Protocols.
Embedded Software (ES)	The software embedded in the Smart Card Integrated Circuit. The ES may be in any part of the non-volatile memories of the Smart Card IC.
Embedded software developer	Organisation responsible for the Smart Card embedded software development and the specification of pre-personalisation requirements.
Initialisation	The process of writing specific information into Non-Volatile Memory during IC manufacturing and testing as well as executing security protection procedures by the IC manufacturer. The information may contain protection codes or cryptographic keys.
Initialisation Data	Specific information written during manufacturing or testing of the TOE.
Integrated Circuit (IC)	Electronic component(s) designed to perform processing and/or memory functions.
IC designer	Organisation responsible for the IC development.

IC manufacturer	Organisation responsible for the IC manufacturing, testing, and pre-personalisation.
IC packaging manufacturer	Organisation responsible for the IC packaging and testing.
Master File (MF)	As defined in 1.6ISO/IEC 7816-3, Identification Cards-Integrated circuit(s) cards with contacts-Part 3: Electronic Signals and Transmission Protocols.
Message Authentication Code (MAC)	A MAC is a message digest appended to the message itself. The MAC cannot be computed or verified unless a secret is known. It is appended by the sender and verified by the receiver who is then able to detect a message falsification. ITSO MACs are derived based on all (rather than selected) fields of the message they are accompanying.
Non-volatile memory	A semiconductor memory that retains its content when power is removed (i. e. ROM, EEPROM, FLASH).
Personalisation	The process of writing specific information into the non-volatile memory in preparing the IC for issuance to users.
Personaliser	Organisation responsible for the Smart Card personalisation and final testing.
Photomask	A mask that is used during chip manufacturing to protect selected parts of a silicon wafer from a light source while allowing other parts of the surface of the wafer to be exposed. The photomask is the means by which the chip's circuits, and therefore its functionality, are placed on the chip.
Platform	An operational smart card system.
Simple Power Analysis (SPA)	A technique in which physical measurements of power consumption over time are used to identify IC operating details. SPA can, in some instances, provide information leading to recovery of internal operational parameters, keys, etc.
Smart Card product manufacturer	Organisation responsible for the Smart Card product Finishing process and testing.
Smart Card Application Software (AS)	The part of ES dedicated to the applications.

9.3 ITSO Terminology

This section contains only those ITSO specific terms that are used in this Security Target.

Cyclic Redundancy Check (CRC)	A checksum calculation added as a field to allow a check on the integrity of preceding data.
Host Operator or Processing System (HOPS)	A conceptualised 'back office' facility which represents various scales of implementation from a small PC to a complex processing facility at a large 'clearing' center.
Integrated Transport Smartcard Organisation (ITSO)	A public-private partnership of major bus and rail transport groups, smaller public transport operators, Passenger Transport Executives, Transport for London and Shire Counties plus the rail industry represented collectively by the Association of Train Operating Companies (ATOC).
ITSO HOPS Security Module (IHOPS)	A 'back-office' security access module (SAM) that may undertake card management responsibilities (HOPSAMS) or key management tasks (HOPSISMS) for the ITSO Scheme.
ITSO Interoperable Product Entity (IPE)	The representation of a product (ticket) as held within the ITSO shell, defined in terms of standard data elements.
ITSO Security Module (ISAM)	An ITSO SAM
ITSO Shell	The ITSO application loaded onto a single or multi application smart card.
ITSO Specification	A standardised method of describing transport tickets and other transport products stored electronically in a smart card or other device, processing transactions associated with those tickets and passing information relating to their sale or use to other organisations that are members of ITSO. The objective of this standardisation is to define a platform and tool-box for the provision of interoperable contactless smart card public transport ticketing and related services in the UK in a manner which offers end to end loss-less data transmission and security.
Multi-function Operating System (Mfos)	A smartcard operating system developed by Ecebs Ltd.
Point of Service Terminal (POST)	A terminal where the shell is read/written to, as appropriate, to add ITSO products or travel rights, to check the validity of ITSO products or to modify/remove ITSO products and or travel rights.

Security Access Module (SAM)	A hardware device built into every POST, which allows the terminal to identify itself and to hold information in secure conditions. A critical component of the ITSO security architecture.
Scheme	A particular implementation of a commercial agreement between participants resulting in a smart card implementation (the issue and acceptance of a product or group of products).
Transaction	The complete process from when a card with an ITSO Application loaded on it is first detected and processing commences until a record is made of the event.
Transaction Sequence Number	A three byte binary number which when combined with the identity of SAM that created it forms a unique number for each transaction record authenticated by that SAM.