APACS

PIN ENTRY DEVICE PROTECTION PROFILE

Association for Payment Clearing Services
Mercury House, Triton Court,
14, Finsbury Square
LONDON. EC2A. 1LQ

Telephone 020 7711 6200
Facsimile 020 7628 0924
Website www.apacs.org.uk

Produced for: **The Card Payments Group**
Creation Date: **July, 2003**

Version No.: **1.37**

Requests for such permission should be addressed to:

The Manager,
Technical Standards & ATMs Section,
APACS, Card Services,
Mercury House, Triton Court,
14, Finsbury Square,
London. EC2A 1LQ

Any enquiries regarding the content of the document please e: mail trevor.day@apacs.org.uk

TABLE OF CONTENTS

REFERENCES

ANSI X9.17 Financial Institution Key Management (Wholesale), 1985, Appendix C Pseudo Number Generation.

CC Common Criteria for Information Technology Security Evaluation (Comprising Parts 1-3, [CC1], [CC2], [CC3]).

CC1 Common Criteria for Information Technology Security Evaluation
Part 1: Introduction and General Model
CCIMB-99-031, Version 2.1, August 1999.

CC2 Common Criteria for Information Technology Security Evaluation
Part 2: Security Functional Requirements
CCIMB-99-032, Version 2.1, August 1999.

CC3 Common Criteria for Information Technology Security Evaluation
Part 3: Security Assurance Requirements
CCIMB-99-033, Version 2.1, August 1999.

CEM Common Methodology for Information Technology Security Evaluation
Part 2: Evaluation Methodology,
CEM-99/045, Version 1.0, August 1999.

EBS EBS 105-1: PIN BASED POS Systems Part One Minimum Criteria for Certification Procedures, Version 2, December 1998.

EBS 105-2: PIN BASED POS Systems Part Two POS Systems with Online PIN Verification: Minimum Security and Evaluation Criteria, Version 2, December 1998.

EBS 105-3: PIN BASED POS Systems Part Three POS Systems with Offline PIN Verification: Minimum Security and Evaluation Criteria, Version 2, December 1998.

FIPS 46-3 Data Encryption Standard DES.

FIPS 197 Advanced Encryption Standard AES.

ISO 9564-1 Banking - Personal Identification Number (PIN) management and security. Part-1: Basic principles and requirements for online PIN handling for ATM and POS systems.

ISO 11568 ISO 11568-1:1994 Banking -- Key management (retail) -- Part 1: Introduction to key management.

ISO 11568-2:1994 Banking -- Key management (retail) -- Part 2: Key management techniques for symmetric ciphers.

ISO 11568-3:1994 Banking -- Key management (retail) -- Part 3: Key life cycle for symmetric ciphers.

ISO 11568-4:1998 Banking -- Key management (retail) -- Part 4: Key management techniques using public key cryptosystems.

ISO 11568-5:1998 Banking -- Key management (retail) -- Part 5: Key life cycle for public key cryptosystems.

ISO 11568-6:1998 Banking -- Key management (retail) -- Part 6: Key management schemes.

EMV EMV2000 Integrated Circuit Card Specification for Payment Systems
Book 1 - Application Independent ICC to Terminal Interface Requirements Version 4.0 dated Dec 2000.

EMV2000 Integrated Circuit Card Specification for Payment Systems
Book 2 – Security and Key Management Version 4.0 dated Dec 2000.

EMV2000 Integrated Circuit Card Specification for Payment Systems
Book 3 – Application Specification Version 4.0 dated Dec 2000.

EMV2000 Integrated Circuit Card Specification for Payment Systems
Book 4 – Cardholder, Attendant and Acquirer Interface Requirements Version 4.0 dated Dec 2000.

PED APACS PIN Entry Device Guideline 11 Version 1.5.

TSRPP Transactional Smartcard Reader Protection Profile, Version 2.0, Jan 2000.

DOCUMENT VERSION CONTROL:

Version	Author	Comment
0.5	A Chilver	18 Apr 02 - PED Security Evaluation Criteria
0.25	Logica	10 Jun 02 - The PED Security Evaluation Criteria was the primary input to create the first version of the Protection Profile
1.0	Logica	3 Jul 02 – Evaluated Version
1.1	APACS	19 Nov 02 – Pre-Evaluation version amended following peer review by product developers.
1.2	APACS	Final version for Evaluation / Certification
1.3	APACS	Correction to typographical errors: 6.1.2.4 FCS_COP1.1 refers to ISO 9564-1. PIN block formats are specified in 9564 part 2. *Application note* under FIA_AFL.1.2 (2) - first sentence up to [EMV] refers, but the remainder of this note should refer within the application note over the page under 5.1.2.5 FIA_UID.2.1.
1.35	APACS	Minor pre-certification amendments to meet mutual recognition requirements from the UK CB.
1.36	APACS	Final pre-certification amendments to meet mutual recognition requirements from the UK CB
1.37	APACS	Certified Version

1. INTRODUCTION

1.1 PP Identification

Title: PIN Entry Device CC Protection Profile

Author: Trevor Day

Reviewer: Colin Whittaker

Publishing Date: 11 July 2003

Issue Number: 1.37

Sponsoring Organisation: APACS

Version of CC used for development: CC Version 2.1 (also known as ISO 15408).

1.2 PP overview

This protection profile has been developed to identify and describe the basic security requirements needed to protect the PINs, security related critical values and software within PIN Entry Devices where these devices are to be used to provide offline PIN verification. Such offline verification can be used to provide cardholder identification for smartcard based transactions, and such devices may either supplement or replace existing POS terminals, see [PED]. In addition, this protection profile provides segregation between certain classes of application that may be run on these devices.

An increasingly popular method for consumers to pay for goods is via credit or debit cards at the Point Of Sale. Unfortunately, current magnetic stripe card credit and debit cards offer many opportunities for fraud.

To combat these opportunities, it is expected that smartcard based systems with the capability to identify cardholders and to verify cards will increasingly be used. These systems are built around the customer inserting their card into a card reader followed by the entry of an identifying PIN. In order to perform these functions securely, the confidentiality and integrity of the customer's PIN and the verification system must be assured.

The environment for this protection profile consists of a device comprising a PIN entry device with integral display, a smartcard interface device (IFD) and a POS terminal. These three components may be combined to form one to three separate physical units where each unit shares a common physical enclosure.

The TOE encompasses the components of the environment in which a PIN can be entered, processed or reside in a non-enciphered format.

This PP defines a core set of requirements applicable to all such devices and an additional set of requirements, the Local Encryption functional package, applicable to devices where the PIN must be communicated between separate physical units. The core requirements cover functional requirements in the following domains:

- the physical security of the system against probing, tampering and stressing,
- the electromagnetic environment including EM interference and compatibility issues,
- the secure path between the external environment and the TSF,
- the logical security of the embedded software,
- the authorised roles and services.

The Local Encryption functional package covers functional requirements in the following additional domains:

- the security of data communicated between secure components of the TOE,
- cryptographic key management,
- cryptographic algorithms.

This PP also identifies assurance requirements that cover characteristics of the design and the product life-cycle, and the analysis of the vulnerabilities of such systems.

1.3 CC Conformance

This PP is Part 2 extended and Part 3 augmented for EAL4.

1.4 Scope

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

- Section 2 is the TOE Description.
- Section 3 provides the statement of TOE security environment.
- Section 4 provides the statement of security objectives.
- Section 5 provides the statement of the core IT security requirements.
- Section 6 provides the Local Encryption Functional Package.

- Annex A provides the security objectives, security requirements and TOE summary specification rationales.

1.5 Terminology

This section contains definitions of technical terms that are used with a meaning specific to this document. Terms defined in the [CC] are not reiterated here, unless stated otherwise.

Administrator: these are the authorised users who maintain and support the equipment; amongst their tasks may be key management, equipment installation and upgrades.

ANSI: American National Standards Institute.

APACS**:** Association for Payment Clearing Services.

ATM: Automatic Teller Machine.

Automated key distribution**:** the distribution of cryptographic keys, usually in encrypted form, using electronic means such as a computer network (e.g., down-line key loading, the automated key distribution protocols of ANSI X9.17).

CardHolder**:** the rightful user of the card.

Credit card: a card for which a credit agreement is in place.

Cryptographic key (key)**:** a parameter used in conjunction with a cryptographic algorithm that determines, for example:

- the transformation of plain text data into ciphertext data;
- the transformation of ciphertext data into plain text data;
- a digital signature computed from data;
- the verification of a digital signature computed from data;
- a data authentication code (DAC) computed from data.

Debit card**:** a card for which a debit agreement is in place.

DES**:** Data Encryption Standard (see FIPS PUB 113).

Electromagnetic compatibility (EMC)**:** the ability of electronic systems to operate in their intended environments without suffering an unacceptable degradation of the performance as a result of unintentional electromagnetic radiation or response.

Electromagnetic interference (EMI)**:** electromagnetic phenomena which either directly or indirectly can contribute to a degradation in the performance of an electronic system.

Encrypted key (ciphertext key)**:** a cryptographic key that has been encrypted with a key encrypting key, a PIN or a password in order to disguise the value of the underlying plain text key.

FIPS: Federal Information Processing Standard.

Firmware**:** the programs and data (i.e. software) permanently stored in hardware (e.g., in ROM, PROM, or EPROM) such that the programs and data cannot be dynamically written or modified during execution. Programs and data stored in EEPROM are considered software.

Hardware**:** the physical equipment used to process programs and data in a cryptographic module.

IFD**:** smartcard interface device allowing the smartcard to communicate (read/write) to the outside world.

Integrity**:** the property that sensitive data has not been modified or deleted in an unauthorised and undetected manner.

Interface**:** a logical section of a cryptographic module that defines a set of entry or exit points that provide access to the module, including information flow or physical access.

ISO**:** International Organisation for Standardisation.

Key encrypting key: a cryptographic key that is used for the encryption or decryption of other keys.

Key management: the activities involving the handling of cryptographic keys and other related security parameters (e.g. counters) during the entire life cycle of the keys, including their generation, storage, distribution, entry and use, deletion or destruction, and archiving.

Manual key distribution: the distribution of cryptographic keys, often in a plain text form requiring physical protection, but using a non-electronic means, such as a bonded courier.

Manual key entry: the entry of cryptographic keys into the TOE from a printed form, using, for example, the pin pad or a keyboard.

PIN Entry Device: a device for inputting a PIN.

PED: a PIN Entry Device with integral display.

Personal Identification Number (PIN): e.g. a 4 to 12 character alphanumeric code or password used to authenticate an identity, commonly used in banking applications.

Physical protection: the safeguarding by physical means of a module that processes PINs, carries cryptographic keys or other critical security parameters.

PIN: see "Personal Identification Number".

PIN pad: A secure entry device that allows cardholders to key in their PINs in privacy.

Plain text key: an unencrypted cryptographic key, which is used in its current form.

POS: Point of Sale.

Power on/off states**:** states for primary, secondary, or backup power. These states may distinguish between power applied to different portions of the module.

Private key: a cryptographic key used with a public key cryptographic algorithm, uniquely associated with an entity, and not made public.

Program image: the full set of objects (executable code, data, etc.) that are required to perform the whole task(s) for which the program was designed.

PROM: programmable read-only (non-volatile) memory.

Public key: a cryptographic key used with a public key cryptographic algorithm, uniquely associated with an entity, and which may be made public.

Public key certificate: a set of data that unambiguously identifies an entity, contains the entity's public key, and is digitally signed by a trusted party.

Public key (asymmetric) cryptographic algorithm: a cryptographic algorithm that uses two related keys, a public key and a private key; the two keys have the property that, given the public key, it is computationally infeasible to derive the private key.

RAM: Random Access Memory (volatile memory).

ROM: read-only memory (non-volatile memory).

Secret key: a cryptographic key used with a secret key cryptographic algorithm, uniquely associated with one or more entities, and which shall not be made public. The use of the term "secret" in this context does not imply a classification level; rather, it implies the need to protect the key from disclosure or substitution.

Secret key (symmetric) cryptographic algorithm: a cryptographic algorithm that uses a single, secure key for both encryption and decryption.

Self-test states: states for performing self-tests on the module.

Software: the programs, and possibly associated data, that can be dynamically written and modified.

Supervisor: The supervisor is an authorised user that is trained to perform local non-security relevant supervisory functions such as date and time changes, language selection and initialisation.

Trusted path: a mechanism by which a person or process can communicate directly with a secure module and which can only be activated by the person, process or module, and cannot be imitated by untrustworthy software within the module.

1.6 Common Criteria Terminology

Assets: Information or resources to be protected by the countermeasures of a TOE.

Assignment: The specification of an identified parameter in a component.

Assurance: Grounds 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 Part 3 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.

Class: A grouping of families that share a common focus.

Component: The smallest selectable set of elements that may be included in a PP, an ST, or a package.

Connectivity: The property of the TOE, which allows interaction with IT entities external to the TOE. This includes exchange of data by wire or by wireless means, over any distance in any environment or configuration.

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.

Element: An indivisible security requirement.

Evaluation: Assessment of a PP, an ST or a TOE, against defined criteria.

Evaluation Assurance Level (EAL): A package consisting of assurance components from Part 3 that represents a point on the CC predefined assurance scale.

Evaluation authority: A body that implements the CC for a specific community by means of an evaluation scheme and thereby sets the standards and monitors the quality of evaluations conducted by bodies within that community.

Evaluation scheme: The administrative and regulatory framework under which the CC is applied by an evaluation authority within a specific community.

Extension: The addition to an ST or PP of functional requirements not contained in Part 2 and/or assurance requirements not contained in Part 3 of the CC.

External IT entity: Any IT product or system, untrusted or trusted, outside of the TOE that interacts with the TOE.

Family: A grouping of components that share security objectives but may differ in emphasis or rigour.

Formal: Expressed in a restricted syntax language with defined semantics based on well-established mathematical concepts.

Human user: Any person who interacts with the TOE.

Identity: A representation (e.g. a string) uniquely identifying an authorised user, which can either be the full or abbreviated name of that user or a pseudonym.

Informal: Expressed in natural language.

Internal communication channel: A communication channel between separated parts of TOE.

Internal TOE transfer: Communicating data between separated parts of the TOE.

Inter-TSF transfers: Communicating data between the TOE and the security functions of other trusted IT products.

Iteration: The use of a component more than once with varying operations.

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

Organisational security policies: One or more security rules, procedures, practices, or guidelines imposed by an organisation 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.

Product: A package of IT software, firmware and/or hardware, providing functionality designed for use or incorporation within a multiplicity of systems.

Protection Profile (PP): An implementation-independent set of security requirements for a category of TOEs that meet specific consumer needs.

Reference monitor: The concept of an abstract machine that enforces TOE access control policies.

Reference validation mechanism: An implementation of the reference ***monitor concept that possesses the following properties:*** it is tamperproof, always invoked, and simple enough to be subjected to thorough analysis and testing.

Refinement: The addition of details to a component.

Role: A predefined set of rules establishing the allowed interactions between a user and the TOE.

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

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.

Selection: The specification of one or more items from a list in a component.

Semiformal: Expressed in a restricted syntax language with defined semantics.

Strength of Function (SOF): A qualification of a TOE security function expressing the minimum efforts assumed necessary to defeat its expected security behaviour by directly attacking its underlying security mechanisms.

SOF-basic: A level of the TOE strength of function where analysis shows that the function provides adequate protection against casual breach of TOE security by attackers possessing a low attack potential.

SOF-medium: A level of the TOE strength of function where analysis shows that the function provides adequate protection against straightforward or intentional breach of TOE security by attackers possessing a moderate attack potential.

SOF-high: A level of the TOE strength of function where analysis shows that the function provides adequate protection against 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.

System: A specific IT installation, with a particular purpose and operational environment.

Target of Evaluation (TOE): An IT product or system and its associated administrator and user guidance documentation that is the subject of an evaluation.

TOE resource: Anything useable or consumable in the TOE.

TOE Security Functions (TSF): A set consisting of all hardware, software, and firmware of the TOE that must be relied upon for the correct enforcement of the TSP.

TOE Security Functions Interface (TSFI): A set of interfaces, whether interactive (man-machine interface) or programmatic (application programming interface), through which TOE resources are accessed, mediated by the TSF, or information is obtained from the TSF.

TOE Security Policy (TSP): A set of rules that regulate 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: Communicating data to entities not under control of the TSF.

Trusted channel: A means by which a TSF and a remote trusted IT product can communicate with necessary confidence to support the TSP.

Trusted path: A means by which a user and a TSF can communicate with necessary confidence to support the TSP.

TSF data: Data created by and for the TOE that might affect the operation of the TOE.

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 or external IT entity) outside the TOE that interacts with the TOE.

User data: Data created by and for the user that does not affect the operation of the TSF.

2. TOE DESCRIPTION

2.1 Intended Use

A popular method for consumers to pay for goods is via either credit or debit cards at the Point Of Sale. Unfortunately, current magnetic stripe card credit or debit cards offer many opportunities for fraud.

To combat these opportunities, it is expected that smartcard based systems with the capability to identify cardholders and to verify cards will increasingly be used. These systems are built around the customer inserting their card into an interface device (IFD) or card reader, and the entry of an authenticating personal identification number (PIN). In order to perform these functions securely, the confidentiality and integrity of the customer's PIN, various security attributes of the system and the verification system must be assured.

The Target of Evaluation (TOE) for this protection profile encompasses the components of the environment in which a PIN can be entered, processed or reside in a non-enciphered format. These components may include some or all of these components**:** a PIN Entry Device with integral display (PED); a smartcard interface device (IFD) a POS terminal.

These three components may be combined in five different ways giving a number of components, which share the same physical enclosure. They may be formed into one; two or three separate but connected physical units, thus:

CLASS	HARDWARE CONFIGURATION
A	(PED and IFD combined) connected to Terminal.
B1	(PED) connected to (Terminal and IFD combined).
B2	(PED) connected to (Terminal) connected to (IFD).
C	(PED and Terminal and IFD combined).
D	(PED and Terminal combined) connected to (IFD).

Table 1 - Classes of TOE Device

In order that the identity of the cardholder may be authenticated, the PIN must be presented to the smartcard. For class A or C devices, encryption may be unnecessary to assure the continuing confidentiality and integrity of the PIN because the protection afforded by the common physical enclosure surrounding the PED and the IFD may be sufficient.

However, if in order to pass the PIN from the PED to the IFD, the PIN must be passed outside a physical unit as in the remaining classes, then the PIN will need to be encrypted to prevent modification or capture, (see example in Figure 1).

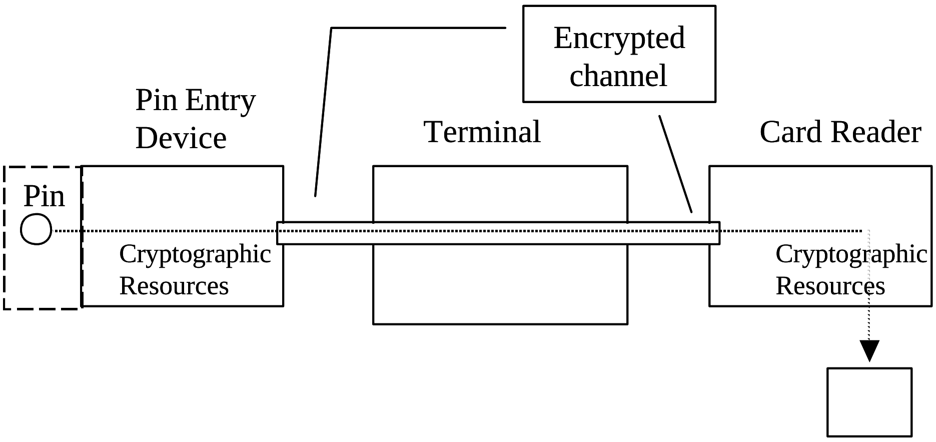

Figure 1 – Class B2 device

In those cases, where smartcards are able to support asymmetric key processing, the PIN may be encrypted at the PED with the public key of the smartcard for forward transmission to the card. As an alternative in this case, the PIN may be encrypted using symmetric key processing until it has reached the IFD, and then decrypted, and then re-encrypted with the public key of the smartcard.

Where smartcards are unable to support asymmetric key processing, the PIN is encrypted using symmetric key processing until it has reached the IFD, and then decrypted and then passed in clear to the smartcard, the physical enclosure of the IFD protecting its integrity and confidentiality.

2.2 Security Features

Attacks on these devices comprise:

- attacks on the PIN which may be either to its integrity or confidentiality;
- attacks on the cryptographic mechanisms of the TOE, such as, the secret keys or key generation seeds;
- attacks on the authenticated applications of the TOE.

The countermeasures that this PP identifies include:

- Physical constraints in the environment of the TOE to prevent the interception of PINs, either by visual or auditory means, on input to the TOE;
- Physical tamper protection and detection of the TOE;
- Optionally, the protection of PINs and other critical data by encryption when outside the secure physical containment;
- Management and protection of cryptographic keys used to ensure the confidentiality or integrity of PINs input to the TOE;
- Segregation of authenticated and unauthenticated applications.

The scope of this Protection Profile (PP) is similar to standards such as the European Committee for Banking Standards (ECBS) [EBS] and related protection profiles such as the Transactional Smartcard Reader Protection Profile [TSRPP].

3. TOE SECURITY ENVIRONMENT

3.1 Introduction

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

To this end, the statement of TOE security environment identifies and lists the assumptions made on the environment and the intended method of use of the TOE, defines the threats that the TOE is designed to counter, and the organisational security policies with which the TOE is designed to comply.

3.2 Environmental and Method of Use Assumptions

This section describes the assumptions about the environment in which the TOE is to be used and its intended method of use.

[A.No_Evil] It is assumed that there are one or more individuals, the administrators, who are assigned to maintain and support the TOE in its operational environment and that these individuals are not careless, wilfully negligent or hostile.

3.3 Assumed Threats

This section describes the threats to the assets that require protection.

3.3.1 Assets

These devices are intended to be used to perform offline PIN verification, that is verification in potentially hostile user environments where these devices are not under constant scrutiny.

The primary assets of concern to this PP are the PINs of users, that is customers, wishing to authenticate themselves, and the confidentiality of the information associated with authenticated applications. A PIN derives its value from the potential financial loss a customer might incur as a result of its compromise, and also the impact such a compromise might have on the reputation of the banking authorities.

Secondary assets whose confidentiality and integrity must be protected consist of characteristics of the TOE important for the security of the system. These assets include:

- cryptographic keys used by the security processes of the TOE;
- random or pseudo random numbers used as nonces within the system;
- the software design and implementation upon which the security of the TOE relies.

3.3.2 Threat Agents

The threat agents can be categorised as:

- authorised users of the TOE (those users who have some authorisation to use the TOE, or exercise supervisory or administrative functions);
- unauthorised users of the TOE.

When the threat may be come from either authorised or unauthorised users these are simply called attackers. Authorised users may perform in various day-to-day roles: ordinary users, supervisors, administrators, etc. Administrators are not considered threat agents see 3.2 **[A.No_Evil]**.

Attackers are assumed to have various levels of expertise, motivation and resources. Expertise could be in emanations (acoustic or EM radiation) gathering, software engineering, the TOE itself or hacking. Their motivation would most likely arise from economic reward. Resources may range from personal computers to sophisticated detection, test and measurement equipment.

3.3.3 Threats

The TOE may be subject to a number of threats against the confidentiality and integrity of its data, software, and services. The attacks may be against the physical and logical characteristics of the TOE.

A user may try to access any elements of the TOE, for which they have no authorisation via some sequence of inputs to the TOE, or by trying to gain a service for which they are not authorised.

> **[T.Manipulation]** An attacker may try to gain access to services or information protected by the TOE for which he is not authorised.

An unauthorised user may try to modify any elements of the TOE, or authorised users may try, for example, to modify program images, cryptographic parameters, or other critical security parameters of the TOE for which they have no authorisation. Physical modifications to the TOE are considered under T.Penetration.

> **[T.Modification]** An attacker may try to modify services or information protected by the TOE for which he is not authorised.

An attacker may attempt to ascertain the internal physical representation of the TOE by looking inside the enclosures of the TOE. The goal of the attack would be to identify aspects of the hardware and software security design, and to infer parameters and initialisation data such as PINs, cryptographic keys and identification data which might be available on internal data paths or in registers.

> **[T.Monitoring]** An attacker may try to use passive measures to probe the TOE to reveal design or operational content.

An attacker may subject the TOE or components of the TOE to physical action, e.g. physical perforation or opening of the device in an effort to compromise the TOE, rather than passively probing.

> **[T.Penetration]** An attacker may try to actively interfere with the TOE to cause the TOE to perform outside of its design or to reveal operational content.

The attacker may subject the TOE to an abnormal environment, e.g. changes to the temperature or voltage or EM radiation, whilst physically probing the TOE for leaked information or in an effort to affect the integrity of information.

> **[T.Stress]** An attacker may try to gain or modify information protected by the TOE for which he is not authorised by subjecting it to environmental stress.

3.4 Organisational Security Policies

The TOE must comply with the following organisational security policies:

[P.Crypto] The cryptographic key management, key operations and algorithms used by the TOE shall comply with APACS approved guidelines [PED], which identify the relevant existing international standards.

4. SECURITY OBJECTIVES

4.1 Introduction

This section sets out the division of responsibilities for addressing the security problem, defined in section 3, between the TOE and its environment. The security objectives for the TOE form the basis for deriving the detailed security requirements for the TOE, as specified in section 5 of this PP.

4.2 Security Objectives to be met by the TOE

The objectives which are to be met by the TOE are:

[O.Confidentiality] The TOE must provide functionality to protect the confidentiality of critical data (in particular PINs).

[O.Enforcement] The TOE must ensure that the security policies of the TOE are not bypassed.

[O.Failsafe] The TOE shall preserve a secure state in the event of an error or reset.

[O.IA] The TOE shall identify and authenticate a user before allowing access to the TOE and its resources.

[O.Integrity] The TOE must provide functionality to detect the loss of integrity of critical data and software images.

[O.Manage] The TOE must provide functionality, which enables authorised administrators to effectively manage the security functionality of the TOE, and must ensure that only authorised administrators are able to access such functionality.

[O.Path] The TOE must provide users with secure communications to the components of the TSF.

The TOE must prevent attackers from passively probing the device, thus compromising the TOE.

[O.Probe] The TOE shall protect itself and its assets from physical probing.

The TOE must prevent the loss of information from authenticated to unauthenticated applications.

[O.Seg] The TOE shall provide segregation between secure authenticated and unauthenticated applications running under the operating system of the TOE.

The TOE must prevent attackers exploiting to environmental conditions outside the normal range in an effort to compromise the security of the TOE, e.g. exposing it to physical shock or electromagnetic radiation.

[O.Stress] The TOE shall protect itself and its assets from environmental stress.

The TOE must be safeguarded to prevent physical interference with the TOE, e.g. breaking into the enclosing housing of the TOE leaving the assets of the TOE available to inspection or modification.

[O.Tamper] The TOE shall protect itself and its assets from unauthorised physical tampering.

The remaining two security objectives apply when the TOE requires local cryptography to preserve the confidentiality and integrity of the PIN and other critical data being communicated between the distributed secure components of the TOE, that is for class B1,B2, D and E devices. These objectives lead to the Local Encryption functional package requirements as articulated in section 6.

The key management aspects of generation, distribution, entry, output, and destruction, and the key operation of private or authenticated data transfer are subject to APACS approved standards.

[O.Crypto] The TOE must support cryptographic functions in a secure manner in accordance with the rules defined by P.Crypto, the cryptographic key management and algorithm policies of the TOE.

[O.Audit] The TOE must provide the means of recording security relevant events, so as to:

- assist administrators in the detection of misconfiguration of the TOE security features that would leave the TOE susceptible to attack; and
- hold users (with supervisory functions) accountable for any actions they perform that may be relevant to security.

4.3 Security Objectives to be met by the TOE Environment

The security objectives are assertive statements to describe the broader environmental context in which the TOE is operated in a secure manner.

[OE.Admin] Those responsible for the TOE shall establish and implement procedures for training and vetting administrators of the TOE, or training the superviors.

The following objective is needed to ensure that non-technical aspects of the audit function, such as the analysis of the audit data and their retention of an appropriate period, are met.

[OE.Audit] The administrators will ensure that the audit functionality is used and managed effectively.

[OE.Banking_Authority] The Banking authorities will maintain the security of their cryptographic keys, and will ensure that only authentic public key certificates for the banking authorities are loaded to the devices.

[OE.Unique] The Banking authorities will establish and maintain procedures to ensure the unique identification of the secure components of the TOE.

5. SECURITY REQUIREMENTS

5.1 TOE Security Functional Requirements

This section identifies the security functional requirements (SFRs) required of the TOE to meet its security objectives.

The components taken from [CC2] to specify the SFRs are listed in the table below together with an indication of whether the components are *iterated* (indicated by “(*N)” where N identifies the number of iterations) or *refined*.

Assignment and selection operations to be completed by the ST author are indicated using the same notation as used in [CC2]. Partially completed operations are denoted by *italicisation* of the word *assignment* or *selection* (as appropriate). Completed assignment and selection operations are indicated by *italicised text*. Refinements of components are indicated by **emboldened text**.

CLASS	FAMILY	COMPONENT	REFINED?
FDP	FDP_DAU	FDP_DAU.1	
	FDP_IFC	FDP_IFC.1 (*2)	
	FDP_IFF	FDP_IFF.1 (*2)	
FIA	FIA_AFL	FIA_AFL.1 (*2)	Y
	FIA_SOS	FIA_SOS.1	Y
	FIA_UAU	FIA_UAU.2	
		FIA_UAU.7	Y
	FIA_UID	FIA_UID.2	
FMT	FMT_MOF	FMT_MOF.1	
	FMT_MSA	FMT_MSA.1(1)	Y
		FMT_MSA.3(1)	Y
	FMT_MTD	FMT_MTD.1 (*2)	Y
	FMT_SMR	FMT_SMR.1	

CLASS	FAMILY	COMPONENT	REFINED?
FPT	FPT_AMT	FPT_AMT.1	Y
	FPT_PHP	FPT_PHP.1 (*2)	Y
		FPT_PHP.3 (*4)	Y
		FPT_PHP.X	
	FPT_RVM	FPT_RVM.1	
	FPT_SEP	FPT_SEP.1	
	FPT_TST	FPT_TST.1	
FTP	FTP_TRP	FPT_TRP.1	

Table 2 - Security Functional Requirements in the core model

5.1.1 FDP - User Data Protection

5.1.1.1 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 *the authenticated applications within the secure components of the TOE.*

FDP_DAU.1.2 The TSF shall provide [assignment: *list of subject*s] with the ability to verify evidence of the validity of the indicated information.

Application note: The assignment operation is left for the ST author to complete by specifying who can authenticate the applications. The method of authentication may be, for example, by data authentication code or digital signature. FDP_DAU.2 may be included in the ST to specify the use of digital signatures; as this is hierarchic to FDP_DAU.1, the PP requirements will be satisfied.

This SFR together with FDP_IFC.1(2), FDP_IFF.1(2), FMT_MSA.1(1) and FMT.MSA.3(1) form the Application control policy. FMT_SMR.1 should identify the various roles that are needed.

5.1.1.2 FDP_IFC.1 - Subset information flow control

FDP_IFC.1.1(1) The TSF shall enforce the *key containment control policy* on *input and output interfaces, data, and operations, which cause data to be transferred via input and output interfaces*.

Application note: The 'subjects' of this policy are in fact the entities attempting to use the interfaces of the TOE, through which information may flow.

FDP_IFC.1.1(2) The TSF shall enforce the *Application control policy* on *subjects arising from applications of the operating system, their information and operations*.

Application note: The application authentication requirement is covered by FDP_DAU.1, and forms part of this policy.

5.1.1.3 FDP_IFF.1 - Simple security attributes

FDP_IFF.1.1(1) The TSF shall enforce the *key containment control policy* based on the following types of subject and information security attributes:

a) *type of interface (input/output);*

b) *type of data and encrypted status.*

FDP_IFF.1.2(1) The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rule holds:

a) *keys may only be output if they are public keys or encrypted.*

Application note: This is the "key containment control policy".

FDP_IFF.1.3(1) The TSF shall enforce *no additional information flow control SFP rules*.

Application note: The SFR has been refined by deletion of the word 'the' for clarity.

FDP_IFF.1.4(1) The TSF shall provide *no additional SFP capabilities.*

Application note: The SFR has been refined by deletion of the words 'the following' for clarity.

FDP_IFF.1.5(1) The TSF shall explicitly authorise an information flow based on the following rules: *none.*

FDP_IFF.1.6(1) The TSF shall explicitly deny an information flow based on the following rules: *all data output via the data output interface shall be inhibited whenever an error state exists and during self-tests.*

Application note: FDP_IFF.1 normally has a dependency on FMT_MSA.3. There are no modifiable security attributes associated with this particular information flow that are under the control of role, so this dependency has not been instantiated.

FDP_IFF.1.1(2) The TSF shall enforce the *Application control policy* based on the following types of subject and information security attributes:

a) *Subjects arising from applications of the operating system;*

b) *The authentication status of the associated application (authenticated /unauthenticated).*

FDP_IFF.1.2(2) The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold:

a) *Only subjects and information arising from authenticated applications may be loaded onto secure components of the TOE;*

b) *The subject and information have the same authentication status.*

Application note: *The segregation between authenticated and un-authenticated applications may be achieved in a number of ways: physical separation between the components of the TOE where applications run on separated components of the TOE; separation in time for those operating systems that can only support a single application at one time; and via mechanisms within the operating system for multi-application operating systems.*

FDP_IFF.1.3(2) The TSF shall enforce *no additional information flow control SFP rules*.

Application note: *The SFR has been refined by deletion of the word 'the' for clarity.*

FDP_IFF.1.4(2) The TSF shall provide *no additional SFP capabilities.*

Application note: *The SFR has been refined by deletion of the words 'the following' for clarity.*

FDP_IFF.1.5(2) The TSF shall explicitly authorise an information flow based on the following rules: *none.*

FDP_IFF.1.6(2) The TSF shall explicitly deny an information flow based on the following rules: *none.*

5.1.2 FIA - Identification and authentication

5.1.2.1 FIA_AFL.1 - Authentication failure handling

FIA_AFL.1.1(1) The TSF shall detect when [assignment: number] unsuccessful authentication attempts occur related to *PIN authentication*.

FIA_AFL.1.1(2) The TSF shall detect when [assignment: number] unsuccessful authentication attempts occur related to *supervisor or administrator authentication*.

FIA_AFL.1.2(1) When the defined number of unsuccessful **PIN** authentication attempts has been met or surpassed, the TSF shall [assignment: list of actions].

FIA_AFL.1.2(2) When the defined number of unsuccessful authentication attempts by the supervisor or administrator has been met or surpassed, the TSF shall [assignment: list of actions].

Application note: *The smartcard provides the information on PIN attempts to the TOE and the TOE must respond appropriately. [EMV]*

5.1.2.2 FIA_SOS.1 - Specification of secrets

FIA_SOS.1.1 The TSF shall provide a mechanism to verify that **PINs** meet *the following criteria:*

a) *PINs shall be variable length from 4 to 12 digits to comply with EMV specifications [EMV] and international standards [ISO9564];*

b) *On entry, PINs may have been corrected via the use of a cancel key, and shall be terminated by a validation key.*

5.1.2.3 FIA_UAU.2 - User authentication before any action

FIA_UAU.2.1 The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

Application note: Different authorised users may be authenticate for different purposes.

5.1.2.4 FIA_UAU.7 - Protected authentication feedback

FIA_UAU.7.1 The TSF shall provide only *acoustic and/or visible signals, independent of the key being pressed,* to the user while the **PIN** authentication is in progress.

5.1.2.5 FIA_UID.2 - User identification before any action

FIA_UID.2.1 The TSF shall require that card authentication is successfully completed before allowing any other TSF-mediated actions on behalf of that user.

Application note: In this sense the Users who exercise TSF mediated actions in this context are those who exercise supervisory or administrative functions. The authentication of supervisor or administrator users are left for the ST author to complete by specifying the number of unsuccessful attempts and the actions that should be taken when such events occur. These assignments should be commensurate with the claimed SOF.

5.1.3 FMT - Security management

5.1.3.1 FMT_MOF.1 - Management of security functions behaviour

FMT_MOF.1.1 The TSF shall restrict the ability to *modify the behaviour of* the functions:

a) *those concerned with detection of out of range physical operating conditions;*

b) *[assignment: those concerned with physical tampering with the secure components of the TOE].*

to [assignment: the authorised identified roles].

Application note: If the detection of tampering is via non-IT means, then b) arising as a dependency from FPT_PHP.(2) may be argued away.

5.1.3.2 FMT_MSA.1 - Management of security attributes

FMT_MSA.1.1(1) The TSF shall enforce *Application control policy* to restrict the ability to modify the **authentication status of applications** to [assignment: the authorised identified roles].

5.1.3.3 FMT_MSA.3 - Static attribute initialisation

FMT_MSA.3.1(1) The TSF shall enforce the *Application control policy* to provide *restrictive* default values for the **authentication status of applications** that are used to enforce the SFP.

FMT_MSA.3.2(1) The TSF shall allow the [assignment: the authorised identified roles] to specify alternative initial values to override the default values when an object or information is created.

5.1.3.4 FMT_MTD.1 - Management of TSF data

FMT_MTD.1.1(1) The TSF shall restrict the ability to *assign* the *identifiers of secure components* to *[assignment: identified roles authorised to assign identifiers of secure components within the card accepting scheme].*

FMT_MTD.1.1(2) The TSF shall **prevent** the ability to *assign* the *identifiers of secure components* to *any user.*

5.1.3.5 FMT_SMR.1 - Security management roles

FMT_SMR.1.1 The TSF shall maintain the roles:

a) *the issuer of secure component identifiers within the card accepting scheme*

b) *[assignment: other authorised identified roles].*

Application note: This requirement should reflect the roles that are needed, in particular for: the modification of out of range conditions, the modification of physical tampering conditions, the modification of authentication status of applications, the setting of alternative initial values for application control policy, the assignment of identifiers to secure components in the card accepting scheme, the modification of cryptographic attributes, the setting of alternate initial security attributes for symmetric Cryptography, the setting of expiration times for PIN encrypting keys.

FMT_SMR.1.2 The TSF shall be able to associate users with roles.

5.1.4 FPT - Protection of the TOE Security Functions

5.1.4.1 FPT_AMT.1 - Underlying abstract machine test

FPT_AMT.1.1 The TSF shall run a suite of tests *during initial start-up* to demonstrate the correct operation of the **hardware and firmware**

Application note: This requirement should reflect the need .for the hardware and firmware of the TSF to be tested independently of the applications within the TSF.

5.1.4.2 FPT_PHP.1 - Passive detection of physical attack

FPT_PHP.1.1(1) The TSF shall provide unambiguous detection of **[assignment*: out of range physical operating conditions*]** that might compromise the **TOE**.

FPT_PHP.1.2(1) The TSF shall provide the capability to determine whether **[assignment*: out of range physical operating conditions*]** with the **TOE** has occurred.

FPT_PHP.1.1(2) The TSF shall provide unambiguous detection of physical tampering that might compromise the **TOE**.

FPT_PHP.1.2(2) The TSF shall provide the capability to determine whether physical tampering with the **TOE** has occurred.

Refinement **It shall be highly unlikely that the TOE can be put back into service without any physical tampering being detected.**

Application note: Tampering, Stressing and Probing identify different ways to physically compromise the TOE. Thus, Tampering is defined as the physical breaking in to the housing or enclosure of the TOE, Stressing subjects the TOE to some out of range environmental condition, whilst Probing is the use of any existing openings to investigate the TOE.

5.1.4.3 FPT_PHP.3 - Resistance to physical attack

FPT_PHP.3.1(1) The TSF shall resist *physical tampering* to the *TOE* by responding automatically such that the TSP is not violated.

Refinement: **Automatic response by the TSF shall be at least:**

a) Erasure of the following:

- **any stored master keys,**
- **PIN encrypting keys,**
- **seed values,**
- **PIN values and other related data.**

b) Sufficient memory is to be erased so that subsequent recovery of sensitive data is prevented, and so that all executable code is rendered temporarily inoperable.

Application note: In some circumstances to facilitate rapid execution, it may be enough to delete only critical items such as key encrypting keys and/or master encryption keys.

FPT_PHP.3.1(2) The TSF shall resist *attacks based on the analysis of electromagnetic radiation from the TOE* **by ensuring that numeric values keyed cannot be deduced from such analysis**.

FPT_PHP.3.1(3) The TSF shall resist *physical attacks leading to disclosure or modification* to the *clear text private or secret keys and PINs within the TOE* **by ensuring that cleartext private or secret keys are stored or processed, and that PINS are processed, only in secure components of the TOE.**

FPT_PHP.3.1(4) The TSF shall resist *[assignment: attacks based on unexpected or out of range physical operating conditions]* to the *TOE* by responding automatically so that the TSP is not violated.

Application note: *Such attacks might rely on over or under voltage, or extreme temperatures. The refinement under FPT_PHP.3.1(1) equally applies to this SFR.*

5.1.4.4 FPT_PHP.X.1 - Detection or resistance to physical attack

FPT_PHP.X.1 The TSF shall resist physical attacks based on addition of any PIN tapping device to the PIN Entry Device and Card Reader by [selection: providing the capability to detect such attacks with a high probability, automatically responding such that the TSP is not violated].

Application note: *This functional requirement is an extended component that allows the ST author to select either or both of the options of detecting or automatically responding to this type of physical attack on the TOE. Selecting one of the options leads to an SFR that is equivalent to either FPT_PHP.1 (detection) or FPT_PHP.3 (automatic response). Where the option of 'automatically responding such that the TSP is not violated' is selected, the method of responding must meet the requirements of the refinement made to FPT_PHP.3.1(1) above.*

FPT_PHP.X.1 applies to those physical tampering attacks based on a PIN tapping device, whilst FPT_PHP.1(2) and FPT_PHP.3(1) are required only in respect of other physical tampering attacks. The effect is thus that the ST author can select requirements equivalent to one or both of FPT_PHP.1(2) and FPT_PHP.3(1) for the attacks based on a PIN tapping device, but is required to include both FPT_PHP.1(2) and FPT_PHP.3(1) in respect of other physical tampering attacks.

5.1.4.5 FPT_RVM.1 - Non-bypassability of the TSP

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.

5.1.4.6 FPT_SEP.1 - 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.

5.1.4.7 FPT_TST.1 - TSF testing

FPT_TST.1.1 The TSF shall run a suite of self tests *during initial start-up, upon request of authorised operator if the TOE consists of non-integrated components,* to demonstrate the correct operation of the TSF.

Application note: The initial start-up tests shall include tests to verify the integrity of keys, data and applications that utilise these keys and data of both the integrated components of the TOE and non-integrated components. Non-integrated components are where the PED and/or IFD are not combined with each other or the terminal as set out in the exemplar classes in section 2.1.

FPT_TST.1.2 The TSF shall provide authorised users with the capability to verify the integrity of TSF data.

FPT_TST.1.3 The TSF shall provide authorised users with the capability to verify the integrity of stored TSF executable code.

5.1.5 FTP - Trusted path/channels

5.1.5.1 FTP_TRP.1 - Trusted path

FTP_TRP.1.1 The TSF shall provide a communication path between itself and *users* that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from modification or disclosure.

FTP_TRP.1.2 The TSF shall permit *the TSF* to initiate communication via the trusted path.

FTP_TRP.1.3 The TSF shall require the use of the trusted path *for any data entry via the PIN entry device.*

5.2 TOE Security Assurance Requirements

The target evaluation assurance level for the product is EAL4 augmented (see [CC3] for a definition of EAL4). Additionally, certain assurance requirements elements are refined. For clarity, therefore, the assurance requirements are stated in full below. Note that where the stated refinement restricts only certain aspect of the assurance element, the intent is that other aspects of the unrefined assurance element must also be applied.

CLASS	FAMILY	COMPONENT	REFINED?
ADV	ADV_HLD	ADV_HLD.2	Y
ALC	ALC_LCD	ALC_LCD.1	Y
AVA	AVA_VLA	AVA_VLA.3	Y

Table 3 - Security Assurance Requirements for the model

5.2.1 ADV - Development

5.2.1.1 ADV_HLD.2 - Security enforcing high level design

Developer action elements:

ADV_HLD.2.1D The developer shall provide the high-level design of the TSF.

Content and presentation of evidence elements:

ADV_HLD.2.1C The presentation of the high-level design shall be informal.

ADV_HLD.2.2C The high-level design shall be internally consistent.

ADV_HLD.2.3C The high-level design shall describe the structure of the TSF in terms of subsystems.

ADV_HLD.2.4C The high-level design shall describe the security functionality provided by each subsystem of the TSF.

Refinement: **The high-level design shall describe how the design of the PIN entry device and any requirements on its physical disposition are able to prevent others from observing the PIN value when being entered.**

ADV_HLD.2.5C The high-level design shall identify any underlying hardware, firmware, and/or software required by the TSF with a presentation of the functions provided by the supporting protection mechanisms implemented in that hardware, firmware, or software.

ADV_HLD.2.6C The high-level design shall identify all interfaces to the subsystems of the TSF.

ADV_HLD.2.7C The high-level design shall identify which of the interfaces to the subsystems of the TSF are externally visible.

ADV_HLD.2.8C The high-level design shall describe the purpose and method of use of all interfaces to the subsystems of the TSF, providing details of all effects, exceptions and error messages, as appropriate.

ADV_HLD.2.9C The high-level design shall describe the separation of the TOE into TSP-enforcing and other subsystems.

Evaluator action elements:

ADV_HLD.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_HLD.2.2E The evaluator shall determine that the high-level design is an accurate and complete instantiation of the TOE security functional requirements.

5.2.2 ALC - Life-cycle support

5.2.2.1 ALC_LCD.1 - Developer defined life-cycle model

Developer action elements:

ALC_LCD.1.1D The developer shall establish a life-cycle model to be used in the development and maintenance of the TOE.

ALC_LCD.1.2D The developer shall provide life-cycle definition documentation.

Content and presentation of evidence elements:

ALC_LCD.1.1C The life-cycle definition documentation shall describe the model used to develop and maintain the TOE.

ALC_LCD.1.2C The life-cycle model shall provide for the necessary control over the development and maintenance of the TOE.

Refinement: **The life-cycle definition shall provide sufficient control to prohibit the inclusion of functional trapdoors.**

Evaluator action elements:

ALC_LCD.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.2.3 AVA - Vulnerability assessment

5.2.3.1 AVA_VLA.3 - Moderately resistant

Developer action elements:

AVA_VLA.3.1D The developer shall perform and document an analysis of the TOE deliverables searching for ways in which a user can violate the TSP.

AVA_VLA.3.2D The developer shall document the disposition of identified vulnerabilities.

Content and presentation of evidence elements:

AVA_VLA.3.1C The documentation shall show, for all identified vulnerabilities, that the vulnerability cannot be exploited in the intended environment for the TOE.

Refinement: **The documentation of identified vulnerabilities shall:**

a) include all logical error conditions that might facilitate attempts to compromise assets in the device;

b) demonstrate that the design and related physical disposition of the PIN Entry Device is able to prevent others from observing entry of the PIN value.

AVA_VLA.3.2C The documentation shall justify that the TOE, with the identified vulnerabilities, is resistant to obvious penetration attacks.

AVA_VLA.3.3C The evidence shall show that the search for vulnerabilities is systematic.

Evaluator action elements:

AVA_VLA.3.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_VLA.3.2E The evaluator shall conduct penetration testing, building on the developer vulnerability analysis, to ensure the identified vulnerabilities have been addressed.

AVA_VLA.3.3E The evaluator shall perform an independent vulnerability analysis.

Refinement: The evaluator's search for vulnerabilities shall confirm that the secure components of the TOE support no unspecified functions.

AVA_VLA.3.4E The evaluator shall perform independent penetration testing, based on the independent vulnerability analysis, to determine the exploitability of additional identified vulnerabilities in the intended environment.

AVA_VLA.3.5E The evaluator shall determine that the TOE is resistant to penetration attacks performed by an attacker possessing a moderate attack potential.

5.3 Strength of Function

The claimed strength of function is *SOF-medium*.

The strength of cryptographic algorithms is outside the scope of the CC, and hence the assessment of algorithmic strength will not form part of the TOE evaluation. The evaluation will, however, confirm the correct implementation of the specified cryptographic algorithms which (in accordance with P.Crypto) are considered to have appropriate strength for the intended use.

6. LOCAL ENCRYPTION FUNCTIONAL PACKAGE

The PP requirements should be supplemented with the following SFRs when local symmetric encryption is required to provide added protection for the assets of the TOE: classes B1, B2, and D in section 2. These SFRs are needed to ensure that O.Crypto and O.Audit are upheld, and augment O.Confidentiality, O.Manage, O.Path and O.Tamper.

Note that iteration numbers are continued from the sequences in section 5. The identified iteration numbers in the table below indicate the *total* number of iterations of that component when the core requirements are taken into account.

CLASS	FAMILY	COMPONENT	REFINED?
FAU	FAU_GEN	FAU_GEN.1	
FCS	FCS_CKM	FCS_CKM.1	Y
		FCS_CKM.2	
		FCS_CKM.4	
FCS	FCS_COP	FCS_COP.1	
FDP	FDP_ACC	FDP_ACC.1	
	FDP_ACF	FDP_ACF.1	
FMT	FMT_MSA	FMT_MSA.1(2)	Y
		FMT_MSA.2	Y
		FMT_MSA.3(2)	Y
		FMT_SAE.1	
FPT	FPT_ITT	FPT_ITT.1	Y
	FPT_PHP	FPT_PHP.3(5)	Y
	FPT_STM	FPT_STM.1	
FTP	FTP_ITC	FTP_ITC.1	Y

Table 4 - Security Functional Requirements in the Local Encryption Functional Package

6.1.1 FAU - Security Audit

6.1.1.1 FAU_GEN - Audit data generation

FAU_GEN.1.1 The TSF shall be able to generate an audit record of the following auditable events:

a) Start-up and shutdown of the audit functions;

b) *Events connected with manual changes to cryptographic keys.*

c) [assignment: other specifically defined auditable events].

FAU_GEN.1.2 The TSF shall record within each audit record at least the following information:

a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event;

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [assignment: *other audit relevant informatio*n].

6.1.2 FCS - Cryptographic Services

6.1.2.1 FCS_CKM.1 - Cryptographic key generation

FCS_CKM.1.1 The TSF shall generate **local** cryptographic keys in accordance with a specified cryptographic key generation algorithm [assignment: cryptographic key generation algorithm] and specified cryptographic key sizes [assignment: cryptographic key sizes] that meet the following:

a) *local cryptographic keys shall be generated by using a random or pseudo-process conforming to the ANSI X9.17 standard or equivalent;*

b) *local cryptographic keys shall except, by chance be unique to the TOE, and shall be used for no other purpose than to protect PINs.*

6.1.2.2 FCS_CKM.2 - Cryptographic key distribution

FCS_CKM.2.1 The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method [*assignment: approved cryptographic key distribution method*] that meets the following:

a) *the key establishment procedure between components of the TOE shall meet the ISO 11568 standard.*

b) *symmetric PIN encrypting keys shall be distributed encrypted under a symmetric key encrypting key, or under the public key that corresponds to the private key of the secure component of the TOE.*

6.1.2.3 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 [assignment: cryptographic key destruction method] that meets the following: [assignment: list of standards].

6.1.2.4 FCS_COP.1 - Cryptographic operations

FCS_COP.1.1 The TSF shall perform the *encryption of PINs* in accordance with a specified cryptographic algorithm [*selection: Triple DES, Advanced Encryption Standard, [assignment other approved cryptographic algorithm]*] and cryptographic key sizes [*selection: 112 bits for Triple DES, 128 bits for AES, [assignment: other approved cryptographic key sizes]*] that meet the following: *approved PIN Block Formats conforming to ISO 9564-2 Format 1*, *[selection: FIPS 46-3, FIPS 197], [assignment: list of other approved standards]*.

Application note: APACS Guidelines [PED] identifies the relevant international standards, the current version of the international standards is understood to apply.

6.1.3 FDP - User data protection

6.1.3.1 FDP_ACC.1 - Subset access control

FDP_ACC.1.1 The TSF shall enforce the *Cryptographic access control policy* on [assignment: list of subjects, objects, and operations among subjects and objects covered by the SFP].

Application note: FDP_ACC.1 and FDP_ACF.1 arise as indirect dependencies. A Security Target claiming conformance to this protection profile including the Local Encryption Functional package will need to complete the assignments of this policy to define the access control policy on the cryptographic security attributes of the Security Target identified within the package .

6.1.3.2 FDP_ACF.1 - Security attribute based access control

FDP_ACF.1.1 The TSF shall enforce *the Cryptographic access control policy* to objects based on [*assignment: Cryptographic security attributes, named groups of Cryptographic security attributes*].

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [assignment: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects].

FDP_ACF.1.3 The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [*assignment: rules, based on Cryptographic security attributes, that explicitly authorise access of subjects to objects*].

FDP_ACF.1.4 The TSF shall explicitly deny access of subjects to objects based on the [*assignment: rules, based on Cryptographic security attributes, that explicitly deny access of subjects to objects*].

6.1.4 FMT - Security management

6.1.4.1 FMT_MSA.1 - Management of security attributes

FMT_MSA.1.1(2) The TSF shall enforce *Cryptographic access control policy* to restrict the ability to modify the **Cryptographic** security attributes to [assignment: the authorised identified roles].

6.1.4.2 FMT_MSA.2 - Secure security attributes

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

6.1.4.3 FMT_MSA.3 - Static attribute initialisation

FMT_MSA.3.1(2) The TSF shall enforce the *Cryptographic access control policy* to provide [selection: restrictive, permissive, other property] default values for **Cryptographic** security attributes that are used to enforce the SFP.

FMT_MSA.3.2(2) The TSF shall allow the [assignment: the authorised identified roles] to specify alternative initial values to override the default values when an object or information is created.

Application note: FMT_SMR.1 in the core model will need to identify the authorised identified role that manages the attributes for this instance of FMT.

6.1.4.4 FMT_SAE.1 - Time-limited authorisation

FMT_SAE.1.1 The TSF shall restrict the capability to specify an expiration time for *symmetric PIN encrypting keys* to [assignment: authorised identified roles].

FMT_SAE.1.2 For each of these security attributes, the TSF shall be able to *change the symmetric PIN encrypting* keys after the expiration time for the indicated security attribute has passed.

6.1.5 FPT - Protection of the TSF

6.1.5.1 FPT_ITT.1 - Basic internal TSF data transfer protection

FPT_ITT.1.1 The TSF shall protect **the PIN** from disclosure **by encipherment** when it is transmitted between separate **secure components** of the TOE.

6.1.5.2 FPT_PHP.3 - Resistance to physical attack

FPT_PHP.3.1(5) The TSF shall resist *physical attacks leading to disclosure or modification* to the *symmetric cryptographic functions* of the TOE **by ensuring that such cryptographic functions are only performed in secure components of the TOE.**

6.1.5.3 FPT_STM.1 - Reliable time stamps

FPT_STM.1.1 The TSF shall be able to provide reliable time stamps for its own use.

6.1.6 FTP - Trusted path/channels

6.1.6.1 FTP_ITC.1 - Inter-TSF trusted channel

FTP_ITC.1.1 The TSF shall provide a communication channel between itself **and loading equipment** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.

FTP_ITC.1.2 The TSF shall permit [selection: the TSF, **loading equipment**] to initiate communication via the trusted channel.

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for *the loading of plaintext key components*.

A PP RATIONALE

This annex demonstrates the suitability of the choice of security objectives, security requirements and TOE summary specification aspects.

A.1 Security Objectives Rationale

This section demonstrates how the threats, organisational security policies and assumptions are met by the security objectives. The correlation between the security needs and the objectives is given in table 5, below.

	Objectives:	O.Audit	O.Confidentiality	O.Crypto	O.Enforcement	O.Failsafe	O.IA	O.Integrity	O.Manage	O.Path	O.Probe	O.Seg	O.Stress	O.Tamper	OE.Admin	OE.Audit	OE.Banking_Authority	OE.Unique
Threats																		
T.Manipulation		x	x	x	x	x	x		x	x		x			x	x	x	
T.Modification		x		x	x		x	x	x	x					x	x	x	
T.Monitoring		x		x	x				x	x	x				x	x		
T.Penetration		x		x	x				x	x				x	x	x		x
T.Stress		x		x	x				x	x			x		x	x		
Policies																		
P.Crypto		x		x													x	
Assumptions																		
A.No_Evil															x			

Table 5 - Correlation between the Security Needs and Objectives

A.1.1 Security objectives suitable to counter the threats

The following rationale demonstrates how the objectives counter the threats:

[T.Manipulation] *An attacker may try to gain access to services or information protected by the TOE for which he is not authorised.*

This threat is mainly countered by O.Confidentiality which provides functionality to protect the confidentiality of the critical data and the software images.

O.Crypto supports O.Confidentiality by ensuring that any cryptographic material that supports encryption of the PINs and other critical security attributes are managed in a secure manner throughout the key life-cycle, that is key generation, distribution, access, and destruction.

O.Audit and OE.Audit provide the means of recording security relevant events in particular those in support of O.Crypto.

O.Enforcement ensures that the security policies of the TOE are upheld in particular those relating to the confidentiality of the TOE.

O.Failsafe ensures that the confidentiality of the TOE at the output interfaces is not breached, should the TOE fail or during self-testing.

O.IA supports O.Confidentiality by ensuring that services of the TOE are only available to users who have been identified via their card and authenticated by the input of a correct PIN, that the PIN satisfies the required quality measures, and that it may not be overlooked or eavesdropped.

O.Manage limits the access to management of the TOE to those authorised.

O.Path provides a secure path both to the TSF and between secure components of the TOE. Thus preventing attackers gaining access to services or information of the TOE by compromising the communications paths between users and TSF of the TOE, or separate secure components of the TOE.

O.Seg assists O.Confidentiality by providing segregation between the secure authenticated and the possibly insecure unauthenticated applications that may be running on the underlying operating system of the TOE; segregating those applications that have access to the secure components of the device and those that do not.

OE.Admin ensures that administrators are suitably trained and vetted, and thus use auditing correctly and act so as not to compromise the information at the TOE.

OE.Banking_Authority ensures that the private keys of these authorities are kept secure, and ensures the authenticity of the certificates issued by these authorities.

[T.Modification] *An attacker may try to modify services or information protected by the TOE for which he is not authorised.*

O.Integrity ensures that the integrity of the data and executable code of the TOE may be established at start up. For non-integrated TOEs, the integrity of the keys may also be established at start up and upon request.

O.Crypto ensures that the integrity of the assets of the TOE may be assured when in transmission between the secure components of the TOE.

O.Audit and OE.Audit provide the means of recording security relevant events in particular those in support of O.Crypto.

O.Enforcement ensures that the security policies of the TOE are upheld in particular those relating to the integrity of the TOE.

O.IA supports O.Integrity by ensuring that services of the TOE are only available to users who have been identified via their card and authenticated by the input of a correct PIN, that the PIN satisfies the required quality measures, and that it may not be overlooked or eavesdropped.

O.Manage limits the access to management of the TOE, in particular to those aspects concerned with integrity, to those authorised.

O.Path provides a secure path both to the TSF and between secure components of the TOE. Thus preventing attackers modifying the services or information of the TOE by compromising the communications paths between users and TSF of the TOE, or separate secure components of the TOE.

OE.Admin ensures that administrators are suitably trained and vetted, and thus use auditing correctly and act so as not to compromise the information at the TOE.

OE.Banking_Authority ensures that the private keys of these authorities are kept secure, and ensures the authenticity of the certificates issued by these authorities.

[T.Monitoring] *An attacker may try to perform passive probing of the TOE to reveal design or operational content.*

This threat is mainly countered by O.Probe which ensures that the TOE resists physical attacks that might lead to disclosure of assets of the TOE, in particular, by the analysis of emanations to gain key pad entries or being passed between secure components of the TOE in cleartext.

O.Crypto supports O.Probe by ensuring that any cryptographic material that supports encryption of the PINs and other critical security attributes when in transit between secure components of the TOE, are managed in a secure manner throughout the key life cycle that is key generation, distribution, access, and destruction.

O.Audit and OE.Audit provide the means of recording security relevant events in particular those in support of O.Crypto.

O.Enforcement ensures that the security policies of the TOE are upheld in particular those relating to probing the components of the TOE.

O.Manage ensures that only those authorised may modify the functions concerned with physical tampering behaviours.

O.Path provides a secure path both to the TSF and between secure components of the TOE. Thus preventing attackers gaining access to services or information of the TOE by probing the communications paths between users and TSF of the TOE, or separate secure components of the TOE.

OE.Admin ensures that administrators are suitably trained and vetted, and thus use auditing correctly and act so as not to compromise the information at the TOE.

[T.Penetration] *An attacker may try to actively interfere with the TOE to cause the TOE to perform outside of its design or to reveal operational content.*

This threat is mainly countered by O.Tamper which ensures that the TOE and its assets are protected against physical tampering.

O.Crypto ensures that the cryptographic functions of the TOE are supported in a secure manner in accordance with the cryptographic policy of the TOE, thus assuring the assets of the TOE in transit between the secure components of the TOE.

O.Audit and OE.Audit provide the means of recording security relevant events in particular those in support of O.Crypto.

O.Enforcement ensures that the security policies of the TOE are upheld in particular those relating to tampering with the components of the TOE.

O.Manage ensures among other things, that only those authorised may modify the functions concerned with physical tampering with the TOE, or the cryptographic functions. Moreover the secure components of the TOE are uniquely numbered to resist substitutions.

O.Path provides a secure path both to the TSF and between secure components of the TOE. Thus preventing attackers gaining access to services or information of the TOE by active tampering with paths between users and TSF of the TOE, or separate secure components of the TOE.

OE.Admin ensures that administrators are suitably trained and vetted, and thus use auditing correctly and act so as not to compromise the information at the TOE.

OE.Unique ensures that the secure components of the TOE are uniquely numbered preventing physical substitutions.

[T.Stress] *An attacker may try to gain or modify information protected by the TOE for which he is not authorised by subjecting it to environmental stress.*

This threat is mainly countered by O.Stress which ensures that the TOE is able to respond automatically to attempts to compromise the TOE or its assets by subjecting the TOE to out of range physical conditions, that is environmental stress.

O.Crypto ensures that the cryptographic functions of the TOE are supported in a secure manner in accordance with the cryptographic policy of the TOE, thus assuring the assets of the TOE in transit between the secure components of the TOE.

O.Audit and OE.Audit provide the means of recording security relevant events in particular those in support of O.Crypto.

O.Enforcement ensures that the security policies of the TOE are upheld in particular those relating to stressing the components of the TOE.

O.Manage ensures among other things, that only those authorised may modify the functions concerned with physical out of range behaviours, or the cryptographic functions.

O.Path provides a secure path both to the TSF and between secure components of the TOE. Thus preventing attackers gaining access to services or information of the TOE, whilst subjecting the TOE to environmental stress.

OE.Admin ensures that administrators are suitably trained and vetted, and thus use auditing correctly and act so as not to compromise the information at the TOE.

A.1.2 Security objectives suitable to meet OSPs

The following rationale demonstrates how the objectives achieve the OSPs:

[P.Crypto] *The cryptographic key management, key operations and algorithms used by the TOE shall comply with APACS guidelines [PED], which identify the relevant existing international standards*

O.Crypto and OE.Banking_Authority ensure that the TOE supports cryptographic functions securely and in accordance with the rules defined by P.Crypto.

A.1.3 Security objectives suitable to uphold assumptions

The following rationale demonstrates how the objectives cover the assumptions:

[A.No_Evil] *It is assumed that there are one or more individuals, the administrators, who are assigned to administer maintain and support the TOE in its operational environment and that these individuals are not careless, wilfully negligent or hostile..*

OE.Admin upholds this assumption.

A.2 Security Requirements Rationale

The rationale considers first the objectives for the Core model, and then the additional objectives that are needed for the Local encryption model and the augmentation of the existing O.Confidentiality, O.Manage, O.Path and O.Tamper objectives.

A.2.1 Security Functional Requirements suitable to achieve the security objectives – Core Model

The following table provides the correlation between the security objectives to be met by the TOE in the Core model.

Security Objectives to be met by the TOE	Security Functional Requirement

O.Confidentiality	Subset information flow control
	FDP_IFC.1(1)
	Simple security attributes
	FDP_IFF.1(1)

O.Enforcement	Underlying abstract machine test FPT_AMT.1 Non_bypassability of the TOE FPT_RVM.1 Domain Separation FPT_SEP.1 TSF testing FPT_TST.1
O.Failsafe	Simple security attributes FDP_IFF.1(1)
O.IA	Authentication Failure Handling FIA_AFL.1 Specification of secrets FIA_SOS.1 User authentication before any action FIA_UAU.2 Protected authentication feedback FIA_UAU.7 User identification before any action FIA_UID.2
O.Integrity	TSF testing FPT_TST.1

Security Objectives to be met by the TOE	Security Functional Requirement
O.Manage	**Management of security functions behaviour** FMT_MOF.1 Management of security attributes FMT_MSA.1(1) Static attribute initialisation FMT_MSA.3(1) Management of TSF data FMT_MTD.1(1) Secure TSF data FMT_SMR.1 Security management roles
O.Path	Trusted path FTP_TRP.1

O.Probe	Resistance to physical attack
	FPT_PHP.3(2)
	Resistance to physical attack
	FPT_PHP.3(3)

O.Seg	Basic data authentication FDP_DAU.1 Subset information flow control FDP_IFC.1(2) Simple security attributes FDP_IFF.1.1(2)

O.Stress	Passive detection of physical attack
	FPT_PHP.1(1)
	Resistance to physical attack
	FPT_PHP.3(4)

O.Tamper	Passive detection of physical attack FPT_PHP.1(2) Resistance to physical attack FPT_PHP.3(1) Detection or resistance to physical attack FPT_PHP.X.1

Table 6 - Correlation between Objectives for the TOE and SFRs for the Core model

O.Confidentiality FDP_IFC.1(1) identifies the Key Containment control policy of the TOE and FDP_IFF.1(1) ensures that the policy is enforced that only public keys may be output from the TOE.

O.Enforcement FPT_AMT.1 and FPT_TST.1 ensure that the TSF is initially operating correctly, and that it continues to operate correctly by using a series of tests, in particular of the underlying physical components, and of the data and executable code of the TOE. In addition, where the TOE consists of non- integrated components the encryption keys will also be verified. FPT_RVM.1 ensures that the TSP enforcement functions are invoked and succeed before each function within the TSC proceeds. FPT_SEP.1 ensures that the TSF maintains a separate security domain from untrusted processes.

O.Failsafe FDP_IFF.1(1) ensures that during self tests and when an error state exists no data is output.

O.IA FIA_AFL.1.1 and FIA_AFL.1.2 ensures that an attacker is limited in the number of attempts at guessing a PIN. FIA_AFL.1.3 and FIA_AFL.1.4 ensures the administrator or supervisor is authenticated before being given access to the TSF. FIA_SOS.1 ensures that length of PINs conform to a quality metric, both internally and on entry. FIA_UAU.2 and FIA_UID.2 ensure that no actions may be taken before the smartcard has been inserted and the PIN has been validated. FIA_UAU.7 ensures that the PIN is not visually or audibly disclosed whilst it is being entered by the user.

O.Integrity FPT_TST.1 ensures the integrity of data and executable code. and for non-integrated components, which must maintain the confidentiality of PINs by cryptographic means, the integrity of the cryptographic keys.

O.Manage FMT_MOF.1 ensures that only authorised users may modify the functions concerned with the out of range and the physical tampering behaviours. FMT_MSA.1(1) ensures only authorised users may modify the authentication status of applications, whilst FMT_MSA.3(1) enforces restrictive default values to be given to the authentication status of applications unless overridden. FMT_MTD.1(1) ensures that identifiers for secure components may only be assigned by authorised users within the card accepting scheme, and that they may not be altered. FMT_MTD.3 ensures these identifiers are unique. FMT_SMR.1 ensures that the various roles needed for the correct management of security functions of the TOE exist.

O.Path FTP_TRP.1 ensures that a secure communication path exists for the entry of the PIN data.

O.Probe FPT_PHP.3(2) ensures that the numeric values cannot be deduced from input at the PIN entry device. FPT_PHP.3(3) ensures that clear text private or secret keys are safe from probing by ensuring that they are only held in secure components of the TOE.

O.Seg FDP_DAU.1 provides assurance that authenticated and unauthenticated applications can be identified. FDP_IFC.1(2) identifies the Application control policy for applications on the TOE, whilst FDP_IFF.1.1(2) ensures that information flows may only take place between applications of the same authentication status, moreover that only authenticated applications may reside on the secure components of the TOE.

O.Stress FPT_PHP.1(1) ensures the TOE provides unambiguous detection of out of range physical operating conditions on the secure components of the TOE. FPT_PHP.3(4) ensures an automatic response to out of range physical operating conditions on the secure components of the TOE so that the TSP is not violated.

O.Tamper FPT_PHP.1(2) ensures the TOE provides unambiguous detection of physical tampering that might compromise the secure components of the TOE. FPT_PHP.3(1) ensures that the TOE resists physical tampering to its secure components by automatically responding with a number of actions. FPT_PHP.X.1 ensures that physical attacks based on adding PIN tapping devices are responded to either so that such attacks are detected with a high probability or else automatically so that the TSP is not violated.

A.2.2 Security Functional Requirements suitable to achieve the security objectives – Local Encryption Functional Package

The following table provides the correlation between the additional security objectives that are needed in the Local Encryption context where the PIN must be communicated between separate secure components. Two additional objectives are introduced O.Crypto and O.Audit. Additional security functional requirements are also needed to uphold the objectives O.Confidentiality, O.Manage, O.Path and O.Tamper in this context.

Security Objectives to be met by the TOE	Security Functional Requirement
O.Audit	Audit data generation FAU_GEN.1 Reliable time stamps FPT_STM.1
O.Crypto	Cryptographic key generation FCS_CKM.1 Cryptographic key distribution FCS_CKM.2 Cryptographic key destruction FCS_CKM.4 Cryptographic operations FCS_COP.1

Security Objectives to be met by the TOE	Security Functional Requirement
O.Confidentiality	Subset access control FDP_ACC.1 Security attribute based access control FDP_ACF.1 Basic internal TSF data transfer protection FPT_ITT.1
O.Manage	Management of security attributes FMT_MSA.1(2) Secure security attributes FMT_MSA.2 Static attribute initialisation FMT_MSA.3(2) Time-limited authorisation FMT_SAE.1
O.Path	Inter-TSF trusted channel FTP_ITC.1
O.Tamper	Resistance to physical attack FPT_PHP.3(5)

Table 7 - Correlation between Objectives for the TOE and SFRs for the Local Encryption package

O.Audit FAU_GEN.1 ensures that audit data is generated for security events, in particular for events connected with manual changes to cryptographic keys. FPT_STM.1 ensures that the TSF is able to provide reliable time stamps for the audit records.

O.Crypto FCS_CKM.1 ensures that cryptographic keys are generated using the required properties. FCS_CKM.2 ensures that key establishment and the distribution of symmetric PIN encrypting keys meet the required distribution standards. FCS_CKM.4 ensures that cryptographic keys will be destroyed using procedures that meet the required key destruction standards. FCS_COP.1 ensures that PIN encryption is performed with algorithms that meet approved standards.

O.Confidentiality In addition to those identified in the Core model, the following properties hold:

FDP_ACC.1 identifies the Cryptographic access control policy on the secure security attributes defining the cryptographic elements introduced by this additional functional package. FDP_ACF.1 ensures that this policy is enforced, and defines it characteristics. FPT_ITT.1 ensures that the if the PIN is transferred between separate secure components of the TOE then it enciphered.

O.Manage In addition to those identified in the Core model, the following properties hold:

FMT_MSA.1(2) identifies the Cryptographic access control policy which restricts the authorised users able to modify the cryptographic security attributes associated with the local encryption package. FMT_MSA.2 ensures that only secure values are accepted for Cryptographic security attributes. FMT_MSA.3(2) ensures that the Cryptographic access control policy provides default values for these attributes, and identifies the authorised users that may override these default values. FMT_SAE.1 ensures that an expiration time for symmetric PIN encrypting keys is defined, and identifies the authorised users that may modify this expiration time.

O.Path In addition to those identified in the Core model, the following properties hold:

FTP_ITC.1 ensures that the TSF provides a secure communication channel between itself, and equipment for loading plaintext key components.

O.Tamper In addition to those identified in the Core model, the following properties hold:

FPT_PHP.3(5) ensures that physical attacks leading to disclosure or modification to the symmetric cryptographic functions of the TOE are resisted by performing such cryptographic functions only in secure components of the TOE.

A.2.3 Security Assurance Requirements appropriate

The evaluation assurance level for this PP, namely EAL4 augmented (see [CC3] for a definition of EAL4), is an appropriate level because it is the minimum level that includes those elements of assurance mandated by the APACS standard [PED]. In particular the AVA_VLA.3 component was selected as more appropriate than AVA_VLA.2 (from EAL4) because the latter provides inadequate assurance of protection against physical attack, i.e. it only provides for resistance to attackers with a **low** attack potential.

A.2.4 Strength of Function claims appropriate

The claimed strength of function is *SOF-medium*. This is considered appropriate [CEM, Table B-2] for resistance to an attacker with attack potential of **moderate**.

A.2.5 Security Requirements mutually supportive

A.2.5.1 Requirements are mutually supportive and internally consistent

	FCS_CKM.1	FCS_CKM.2	FCS_CKM.4	FDP_ACC.1	FDP_ACF.1	FDP_IFC.1(1)	FDP_IFC.1(2)	FDP_IFF.1(1)	FDP_IFF.1(2)	FIA_UID.2	FIA_UAU.2	FMT_MOF.1	FMT_MSA.1(1)	FMT_MSA.1(2)	FMT_MSA.2	FMT_MSA.3(1)	FMT_MSA.3(2)	FMT_MTD.1(1)	FMT_SMR.1	FPT_AMT.1	FPT_STM.1
FDP_DAU.1																					
FDP_IFC.1(1)						i		x													
FDP_IFC.1(2)							i		x												
FDP_IFF.1(1)						x		i													
FDP_IFF.1(2)							x		i	i			i			x			i		
FIA_AFL.1										i	x										
FIA_SOS.1																					
FIA_UAU.2										x											
FIA_UAU.7										i	x										
FIA_UID.2																					
FMT_MOF.1										i									x		
FMT_MSA.1(1)									x	i									x		
FMT_MSA.3(1)										i			x						x		
FMT_MTD.1(1)										i									x		
FMT_MTD.1(2)										i									x		
FMT_SMR.1										x											
FPT_AMT.1																					
FPT_PHP.1(1)										i		x							i		
FPT_PHP.1(2)										i		x							i		
FPT_PHP.3(1)																					
FPT_PHP.3(2)																					
FPT_PHP.3(3)																					
FPT_PHP.3(4)																					
FPT_PHP.X																					
FPT_TST.1																				x	
FTP_TRP.1																					

The following table gives the dependencies between the SFRs for the core model.

Table 8 - Dependency matrix for the Core model

Key x - direct dependencies
i - indirect dependencies.

Note: For the Key containment control policy (FDP_IFF.1(1)) the only attributes are inherent properties of data and interfaces, therefore FMT_MSA.1 and FMT_MSA.3 have not been specified, and consequently indirect dependencies for this information flow control policy do not exist either.

All the dependencies (not explained by the above note) are satisfied by the TOE. FIA_UID.2 satisfies the dependencies on FIA_UID.1 as the former is hierarchic to the latter.

	FCS CKM.1	FCS CKM.2	FCS CKM.4	FDP ACC.1	FDP_ACF.1	FDP IFC.1(1)	FDP IFC.1(2)	FDP IFF.1(1)	FDP IFF.1(2)	FIA UID.2	FIA UAU.2	FMT MOF.1	FMT MSA.1(1)	FMT MSA.1(2)	FMT MSA.2	FMT MSA.3(1)	FMT MSA.3(2)	FMT MTD.1(1)	FMT_SMR.1	FPT AMT.1	FPT STM.1
FAU GEN.1																					x
FCS CKM.1		x	x											i	x				i		
FCS CKM.2	x	i	x											i	x				i		
FCS CKM.4	x	i	I											i	x				i		
FCS COP.1	x	i	x											i	x				i		
FDP ACC.1				i	x																
FDP ACF.1				x	i												x				
FMT_MSA.1(2)				x						i									x		
FMT MSA.2				x										x					x		
FMT MSA.3(2)														x					x		
FMT SAE.1																			x		x
FPT ITT.1																					
FPT PHP.3(5)																					
FPT STM.1																					
FTP_ITC.1																					

The additional dependencies for this PP when the core requirements are augmented with the local encryption functional package are given in the following table 9.

Table 9 - Dependency matrix for the Local Encryption model

Key x - direct dependencies
i - indirect dependencies.

All the additional dependencies introduced by the additional requirements are satisfied by the TOE.

A.2.5.1 Justification that the SFRs form a mutually supporting and consistent whole

Whether the Core model, or the Core model plus Local Encryption functional package is considered, the security functional components form a number of separate functional areas which lead to the absence of inconsistency and to a supportive relationship between themselves.

Thus, the trusted path, and identification and authentication requirements control access to the device ensuring that users have an assured communications path from user environment to TSF, and that the services of the TSF are only available authenticated users.

The physical security providing confidentiality and integrity for the assets of the TOE is supported by policies and functions to safeguard the TOE against probing, tampering and environmental stress.

Where the assets of the TOE must be transmitted between secure components, the core model must be augmented with the Local Encryption functional package which manages cryptographic assets securely and provides the capability to encrypt the assets for transmission between the secure components. These cryptographic requirements are supported by audit requirements with effective time stamping which audit changes to the cryptographic elements, and a trusted channel for the loading of any plaintext key components. Management requirements ensure that cryptographic security attributes and the expiration of key encrypting keys are managed in a secure and timely manner, and that these management functions are restricted to the appropriate role.

Enforcement requirements ensure that the security functions of the TOE are tested to be initially correct, and that they are then not subsequently bypassed, whilst the Failsafe requirements ensure the TOE remains secure in the event of an error or reset. Separation requirements ensure domain separation of the TSF and non-interference by untrusted subjects.

Data authentication and information flow policy requirements provide a means to identify particular applications as being “in some sense” trusted, and to separate these applications, which may execute within secure components, from those that may not.

Finally, management requirements in the core model restrict the modification of the out of range conditions, authentication status and unique identification of secure components to the appropriate roles.