Common Criteria for Information Technology Security Evaluation

Visa Smart Card Protection Profile

Draft

Draft Version 1.6

May 4, 1999

This Profile is a draft attempt to set forth and identify a comprehensive list of smart card security requirements based on the draft ISO Standard 15408, the "Common Criteria", (available at http://www.csrc.nist.gov/cc). Other threats and security concerns may arise as technology evolves. Visa is making this Profile available for the benefit of the industry generally, but neither accepts nor assumes any obligation regarding the completeness or effectiveness of the Profile nor any responsibility for updating the Profile as new threats become known.

You are invited to evaluate and review the Profile and provide any comments you may have to Visa by forwarding your comments to Ken Ayer at kayer@visa.com or Lance Johnson at ljohnson@visa.com. Comments or suggestions submitted to Visa will not be held confidential. By submitting comments or suggestions to Visa, you are licensing Visa to incorporate your comments or suggestions into future versions of the Profile without any attribution or payment to you.

The Profile is provided "AS IS," "WHERE IS" AND "WITH ALL FAULTS," without a warranty of any kind. ALL EXPRESS OR IMPLIED REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE HEREBY DISCLAIMED BY VISA.

VISA SHALL NOT BE LIABLE FOR ANY DAMAGES SUFFERED BY LICENSEE OR ANY THIRD PARTY AS A RESULT OF USING THE PROFILE. IN NO EVENT WILL VISA BE LIABLE FOR ANY LOST REVENUE, PROFIT OR DATA, OR FOR DIRECT, INDIRECT, SPECIAL, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF THE USE OF OR INABILITY TO USE THE PROFILE, EVEN IF VISA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

This Protection Profile was prepared with the assistance of Ray-McGovern Technical Consulting, Inc.

This document is paginated from i to iii, from 1 to 80, and from I to III

Table of Contents

TABLE OF CONTENTS **II**

LIST OF TABLES **III**

1 INTRODUCTION **1**

1.1 IDENTIFICATION 1
1.2 OVERVIEW 1

2 TOE DESCRIPTION **5**

2.1 LIFE CYCLE 5
2.2 APPLICATIONS 6
2.3 CRYPTOGRAPHY 7
2.4 ENVIRONMENTS 7

3 TOE SECURITY ENVIRONMENT **8**

3.1 ASSUMPTIONS 8
3.2 THREATS 9
3.3 ORGANIZATIONAL SECURITY POLICIES 17

4 SECURITY OBJECTIVES **19**

4.1 TOE SECURITY OBJECTIVES 19
4.2 ENVIRONMENT SECURITY OBJECTIVES 21

5 IT SECURITY REQUIREMENTS **23**

5.1 TOE IT SECURITY REQUIREMENTS 23
5.2 SECURITY REQUIREMENTS FOR THE IT ENVIRONMENT 50

6 PP APPLICATION NOTES **51**

6.1 ISSUES UNIQUE TO SMART CARDS 51
6.2 SMART CARD SECURITY FUNCTION POLICIES 53
6.3 MANAGEMENT OF FUNCTIONS IN TSF 56

7 RATIONALE **65**

7.1 INTRODUCTION AND TOE DESCRIPTION RATIONALE 65
7.2 SECURITY OBJECTIVES RATIONALE 65
7.3 SECURITY REQUIREMENTS RATIONALE 70
7.4 RATIONALE THAT DEPENDENCIES ARE SATISFIED 75
7.5 RATIONALE FOR STRENGTH OF FUNCTION MEDIUM 79
7.6 RATIONALE FOR ASSURANCE LEVEL EAL4 AUGMENTED 80

ANNEX A -GLOSSARY **I**

List of Tables

Table 5.1 – Security Functional Components 23
Table 5.2 – Security Assurance Components 33
Table 5.3 – Security Requirements for the Environment 50
Table 6.1 – Security Functional Components Management Options 55
Table 7.1 - Environmental Considerations Related to Objectives 65
Table 7.2 - Security Objectives Related to Environmental Considerations 68
Table 7.3 - Security Objectives Related to Security Requirements 70
Table 7.4 - Security Requirements Related to Security Objectives 72
Table 7.5 - Summary of Dependencies 76

1 Introduction

1.1 Identification

Version Number: Draft Version 1.0

Registration:

A glossary of terms used in the Protection Profile (PP) is given in Annex A. This Protection Profile is hereafter referred to as the Visa Smart Card Protection Profile (VSCPP).

This PP has been built with Common Criteria (CC) Version 2.0 and Common Methodology for Information Technology Security Evaluation (CEM) 99/008 Version 0.6 January 1999. .

The structure for this PP was established through use of the Common Criteria Toolbox (Beta Version 2.0, 16 February 1999). This Toolbox was developed by SPARTA, Inc., for the US National Security Agency and is available through them.

A product compliant with this PP may offer security features and functionality beyond that specified in this PP depending on the application.

1.2 Overview

The Target of Evaluation (TOE) is the integrated circuit, operating system, and application(s) of an integrated circuit card, otherwise known as a smart card. This Protection Profile does not include printing, the magnetic stripe, if present, security features such as holograms, or any other part of the card. This Protection Profile also does not apply to the card accepting device (terminal), nor to any network with which the integrated circuit card interfaces.

1.2.1 Definition

“Smart card”as used in this PP means an integrated circuit containing non-volatile memory and a microprocessor, packaged and embedded in a carrier. The integrated circuit typically is a single chip incorporating CPU, RAM, ROM, and programmable nonvolatile memory (usually EEPROM), optionally also including a crypto coprocessor. The carrier is usually made of plastic and usually conforms to ISO 7810 and 7813 - Identification cards - but may have the smaller size of a GSM (Global System for Mobile communications) Subscriber Identification Module (SIM). The chip is embedded in a module that incorporates the communication channels (contact in accordance with ISO 7816 or contactless in accordance with ISO 15443).

1.2.2 Technology

Most smart cards that have been issued use “conventional” smart card technology, in which almost all of the program code is added to the card before it is issued. A new generation of cards is beginning to be deployed that has significantly different characteristics.

1.2.2.1 Conventional Smart Card Technology

A distinction is made between “soft mask” and “hard mask” cards in conventional smart card technology.

A “Soft Mask” card typically has a card operating system (COS) programmed into ROM and application code programmed in programmable nonvolatile memory. This is typically used in pilots, when the code needs to be tested in use and some changes may be anticipated. The flexibility of having the code in programmable nonvolatile memory permits easy “fixes” and “patches,” but this also presents a security risk. Note, however, that a “pilot” may represent a geographically limited but fairly large implementation utilizing many thousands of cards requiring significant security involvement.

A “Hard Mask” card has an operating system tailored to the specific use of the card and the application code stored in ROM. The resultant program code can not be realistically or usefully split into operating system and application; they are merged to become the program. Card cost varies according to the memory on the chip; large-scale deployment requires the lowest cost card possible. Memory is extremely limited, so unnecessary commands are not supported. Virtually all the memory available is used and the ability to add additional code is usually blocked as the final step in personalization. This gives greater security but less flexibility to the program code.

Most conventional smart cards support a single application. However, some smart cards support several applications and this trend is growing. The applications may be related forms of payment (credit, debit, and stored value) or a payment application with some other (“loyalty,” identification, non-financial records, etc.). Such multiple application cards provide new challenges, both in the technological and business aspects.

1.2.2.2 New Generation Smart Card Technology

The newest generation of smart cards has a generally more robust operating system that permits adding or deleting application code after the card is issued. Such cards are generally (in 1999) programmed in Java, Smart Cards for Windows ^{TM}, or MEL (the Multos programming language). Such cards may have a chip operating system, card operating system, and additional layers such as the Visa Open Platform, which offer industry or application specific features. The security requirements for the operating system (and additional layers if present) are more stringent than those for conventional cards. Particular attention must be paid to the procedures for certifying and loading (or deleting) new applications in the field.

It is imperative that Security Targets and actual smart card products be clearly identified as to type of technology in evaluations and that security functions that are present are appropriate to the type of card.

1.2.2.3 Hardware and Software

Many security relevant functionalities can be implemented in hardware or software or a combination of the two. This Protection Profile does not mandate how this functionality is to be implemented. Any Security Target claiming compliance with this Protection Profile should indicate how the required functionality is met.

1.2.3 Assurance Level

The assurance level for this Protection Profile is EAL4 augmented. Augmentation results from the selection of:

AVA_VLA.3 Vulnerability Assessment; Vulnerability Analysis; Moderately resistan tand
ADV_INT.1 Development; TSF internals; Modularity

Strength of Function is Medium.

1.2.4 Related Standards and Documents

ISO 7810 - Identification cards - Physical characteristics

ISO 7813 - Identification cards - Financial transaction cards

ISO 7816 - Identification cards - Integrated circuit cards with contacts

ISO 10202 - Financial Transaction Cards - Security architecture of financial transaction systems using integrated circuit cards

ISO 14443 (Draft) –Contactless Integrated Circuit Cards, Proximity Cards

ISO 15408 –Information Technology –Security Techniques –Evaluation Criteria for IT Security (Hereafter referred to as "Common Criteria")

Common Methodology for Information Security Evaluation (CEM) Version 0.6, 99/008, January 1999.

1.2.5 Related Protection Profiles and Documents

This Protection Profile has evolved from a great deal of work on smart card security conducted over the past decade. Much of this work has been done in conjunction with a variety of organizations (including the Smart Card Forum, Smart Card Industry Association, Eurosmart, more than a dozen commercial evaluation laboratories, and others) and with semiconductor and smart card manufacturers. In particular it has evolved from:

- Visa International Security Guidelines for Chip Architecture and Design, Operating System Design and Vendor Viability, Version 2, November 1997
- Protection Profile 9806 - Smartcard Integrated Circuit (revision of PP 9704)

- Protection Profile 9809 –Smart Card Integrated Circuit with Embedded Software
- Protection Profile 9810 –Smartcard Embedded Software

As in any developing area, there are differences in approach, interpretation, and direction amongst these efforts. It is intended that future documents will reduce such differences leading to a unified understanding and approach to smart card security.

While the assistance provided by these works is gratefully acknowledged, the responsibility for this Protection Profile rests solely with Visa International.

2 TOE Description

2.1 Life Cycle

This Protection Profile recommends the use of the Card Life Cycle stages outlined in ISO 10202, Part 1, which include:

- Manufacture of the IC and ICC
  - IC semi-conductor design and software design
  - IC manufacturing
  - IC assembling
  - IC embedding
- Card Preparation
  - Card Personalization
  - Common Data File (CDF) activation
- Application Data File (ADF) Preparation
  - ADF Allocation
  - ADF Personalization
  - ADF activation
- Card Usage
  - Card use
  - ADF deactivation
  - CDF deactivation
  - CDF reactivation
  - ADF reactivation
- Termination of Use
  - ADF Termination
  - CDF termination
  - Key termination

The definition and meaning of these stages and processes are provided in the standard.

This Protection Profile is primarily oriented to the Card Usage and Card Termination stages of the Life Cycle. However, the threats identified at these stages must be addressed by counter-measures designed and implemented in the Manufacturing and Preparation stages. The Threats and Attacks may occur at any of the life cycle stages.

The terms CDF and ADF, as used above, may not be appropriate to the object-oriented programming used in some "New Generation"cards. In these cards, additional programs may be added during after the cards have been issued, moving some of the Card Preparation activities into the Card Usage stage. As this is a case in which technological development may be ahead of the standard, ISO 10202 is recommended rather than required in this Protection Profile. Software that is embedded in ROM must be provided to the IC manufacturer, who permanently "burns" it in to the chip. Additional software can be added in programmable nonvolatile memory. In conventional cards, this is done at a stage often called "initialization," which is part of the Card Preparation stage. With new generation technology, additional software can be loaded in to programmable nonvolatile memory after the card has been issued. Card issuance is not listed as a separate stage, but it is the first step in Card Usage.

ISO 10202 discusses the Card Issuer as at least potentially distinct from the Application Supplier. With single application cards these are typically the same entity, but with multiple application cards they may be separate.

Any Security Target claiming compliance with this Protection Profile must restate the Card Life Cycle, showing whether and how it conforms to ISO 10202.

2.2 Applications

Typical applications for smart cards in 1999 include:

- Payment
  - Credit
  - Debit
  - Stored Value Purse
  - Stored Token
  - Mass transit - generally dedicated to a single transport system and typically having low value.
- Telephony - Subscriber Identification Module (SIM) for digital mobile telephones.
- Identification - various public and private schemes provide identification credentials to participants. These may be government, corporate, university, or other entities. The identification credentials are typically associated with various rights and duties, defined by the identification provider. These can include membership, driver's licenses, benefit access, passports, national identification, etc. Typically the identification credentials have value in great part because they can not be easily altered by the credential holder, and assets in the credential must be protected against alteration by the cardholder. Digital certificates used in public key systems fit in to this category.
- Secure information storage - e.g., health records, health insurance.
- Loyalty - These are programs like the "Frequent Flyer" points awarded by airlines. Points are added and deleted from the card memory in accordance with program rules. The total value of these points may be quite high and they must be protected against improper alteration in much the same way that currency value is protected.

- Networked applications - smart cards can hold access credentials like passwords that identify a user to a computer network.

Each of these may have somewhat different security requirements and features, roles, and environmental considerations (e.g., whether always used on-line or off-line, usually off-line with the capability of going on-line, etc.).

Some smart cards have single applications, while others have multiple applications. Multiple application cards can be implemented using conventional technology or the new generation of cards with the capability of post-issuance dynamic downloading of new applications. The security requirements for the operating system and procedures for adding or deleting applications are different for these new generation cards.

2.3 Cryptography

A variety of cryptographic keys are typically used with smart cards, including transport keys, personalization keys, application - specific keys, etc. Handling of these keys must be done in accordance with the key management procedures and policies of the Issuing organization.

Cryptography may be implemented in hardware or software, with various algorithms and various key lengths. Many smart cards have dedicated crypto coprocessors that execute DES, triple DES, RSA and other standard algorithms much faster than software implementations can. Some applications use no cryptography, some use private key and some public key systems.

Any TOE claiming compliance with this Protection Profile must handle cryptographic functions in accordance with applicable international, industrial, or organizational policies. This extends to any applications using cryptography, although there may be additional applications on the card that do not use cryptography at all.

2.4 Environments

Smart card environments are highly variable and to some extent application dependent. In general, it is assumed that a smart card is in the uncontrolled possession of the cardholder. The card must therefore protect its assets against unauthorized alteration that may be accomplished with standard personal computers and with laboratory equipment used without any supervision. Typically the cards are designed to be used worldwide in a wide variety of card acceptance devices that may range from parking meters and vending machines to dedicated read/write devices to card readers attached to conventional computers.

3 TOE Security Environment

This section identifies the following:

- Significant assumptions about the TOE's operational environment
- IT-related threats to the organization countered by VSCPP compliant components
- Threats requiring reliance on environmental controls to provide sufficient protection
- Organizational security policies for which VSCPP compliant TOEs are appropriate

3.1 Assumptions

The specific conditions listed below are assumed to exist in the smart card environment.

A.Attack - Attacker Capability

Attackers are assumed to have various levels of expertise, resources and motivation.

Relevant expertise may be in general semi-conductor technology, software engineering, hacking techniques, or the specific TOE. Resources may range from personal computers and inexpensive card reading devices to very expensive and sophisticated engineering test and measurement devices. They may also include software routines, some of which are readily available on the internet. Motivation may include economic reward or the satisfaction and notoriety of defeating expert security.

A.User - User Privilege

Users of the TOE are assumed to possess the necessary privileges to access the information managed by the TOE.

A.Admin - Administrator Competence

It is assumed that one or more authorized administrators are assigned who are competent to manage the security features of the TOE competently and in an on-going basis.

3.2 Threats

VSCPP compliant TOEs are required to counter threats which may be broadly categorized as:

Threats to the TOE

Threats associated with physical attack on the TOE

Threats associated with logical attack on the TOE

Threats associated with inadequate specification

Threats associated with errors in instantiation

Threats associated with unanticipated interactions

Threats regarding cryptographic functions

Threats which monitor information

Miscellaneous threats

Threats to the environment

3.2.1 Threats to the TOE

3.2.1.1 Threats Associated with Physical Attack on the TOE

T.P_Probe - Physical Probing of the IC

An attacker may perform physical probing of the TOE to reveal design information and operational contents.

Such probing may include electrical functions but is referred to here as mechanical since it requires direct contact with the chip internals. Mechanical probing may entail reading data from memory, tapping into data busses, or reading interconnections between functional elements. The goal of the attacker is to identify such design details as hardware security mechanisms, access control mechanisms, authentication systems, data protection systems, memory partitioning, or cryptographic programs. Determination of software design, including initialization data, personalization data, passwords, or cryptographic keys is also a goal.

T.P_Modify - Physical Modification of the IC

An attacker may physically modify the TOE in order to reveal critical design or security related information.

This modification may be achieved through removal of the die from the plastic card, removal of layers in the integrated circuit, and probing for data lines, buses, and memory locations. Modifications may be through cutting traces, adding connections, or modifying circuitry or memory. This may include repairing of blown fuses, re-institution of debug capabilities, or modification of security critical elements of the circuitry through connecting normally secure elements to areas which can be openly accessed. The goal is to identify such design details as hardware security mechanisms, access control mechanisms, authentication systems, data protection systems, memory partitioning, or cryptographic programs. Determination of software design, including initialization data, personalization data, passwords, or cryptographic keys is also a goal.

T.E_Manip - Electrical Manipulation of the IC

An attacker may utilize electrical probing and manipulation of the TOE to modify security critical data so that the TOE can be used fraudulently.

This modification may include manipulation of debug lockouts, first use indicators, card use blocking, blocking function configuration, card block indicators, or card disablement indicators.

3.2.1.2 Threats Associated with Logical Attack on the TOE

T.Us_Error - User Error

An authorized user of the TOE may compromise the security features of the TOE through introduction of false data and inappropriate operations that masquerade as user error.

T.UA_Op - Unauthorized Operations

An attacker may exploit unauthorized operation of the TOE to penetrate or modify the security features of the TOE.

Unauthorized operations may include use of instructions or commands or sequences of commands sent to the TOE in out of normal operations.

T.UA_Load - Unauthorized Program Loading

An attacker may utilize unauthorized programs to penetrate or modify the security functions of the TOE.

Unauthorized programs may include the execution of legitimate programs not intended for use during normal operation or the unauthorized loading of programs specifically targeted at penetration or modification of the security functions.

T.Cmd_Str - Command Manipulation

An attacker may manipulate software commands to reveal memory contents.

This manipulation may be achieved through requests or formats that are out of range or otherwise non-conforming to the "accepted" usage.

T.Forcd_Rst - Forced Reset

An attacker may force the TOE into a non-secure state through inappropriate termination of selected operations.

Attempts to generate a non-secure state in the TOE may be through premature termination of transactions or communications between the TOE and the card reading device, insertion of interrupts, or by selecting related applications that may leave files open.

T.Trns_Integ - Data Transmission Errors

An attacker may use data transmission errors into and out of the TOE to compromise the integrity of security related information.

T.Flt_Ins - Insertion of Faults

An attacker may determine security critical information through insertion of selected data or errors and observing the result.

T.Re-Use - Replay Attack

An unauthorized user may penetrate the TOE through reuse of previously valid authentication data.

T.Load_Mal - Data Loading Malfunction

An attacker may maliciously generate errors in set-up data to compromise the security functions of the TOE.

T.Priv - Abuse by Privileged Users

A careless, willfully negligent, or hostile administrator or other privileged user may create a compromise of the TOE assets through execution of actions, which expose the security functions or the protected data.

3.2.1.3 Threats Associated with Inadequate Specification

T.First_Use - Fraud on First Use

An attacker may gain access to TOE information by unauthorized use of a new, previously unissued TOE.

T.Impers - Impersonation

An attacker may gain access to TOE information by impersonating an authorized user of the TOE.

T.Access - Invalid Access

A user or an attacker of the TOE may access information or resources without having permission from the person who owns or is responsible for the information or resources.

T.Search - Data Space Search

An attacker may utilize a repeated search of the data space to identify critical information.

T.Cmd_Ftn - Inappropriate Command Use

An attacker may exploit the TOE command set to expose memory contents or to change security critical elements in the TOE.

Elements of the command set include specific commands to read, write, or modify data. An attacker may use these or such commands as debug commands to access serial number control, or unauthorized modification to programmable nonvolatile memory, first use indicators, card blocking function configuration, or card block or card disable indicators.

3.2.1.4 Threats Associated with Errors in Instantiation

T.Load_Flt - Data Load Faults

An attacker may exploit errors in set-up to compromise the security functions of the TOE.

Errors could be generated either through simple errors or as the result of failure of some part of the transfer mechanisms. These errors could occur during loading of programs and/or loading of data.

T.SWBld_Fail - Software Build Failures

An attacker may exploit errors in software design to determine sensitive TOE or user data.

Software errors may result in a routine or application failing to perform as required by the design specifications due to software build failures

T.HWBld_Fail - Hardware Build Failures

An attacker may exploit errors in hardware design to determine sensitive TOE or user data.

Hardware errors may result in a routine or application failing to perform as required by the design specifications due to hardware build failures.

3.2.1.5 Threats Associated with Unanticipated Interactions

T.Alt_Ftn - Use of Alternate Functions

An attacker may exploit interactions between life-cycle functions or applications to expose sensitive TOE or user data.

Interactions may include execution of commands which are not required or allowed in the specific application being performed. Examples include use of debug or native COS functions that are unnecessary or that could compromise security. Inappropriate interactions could also include passing of secure information such as PINs or cryptographic data between applications, or the transfer of value or information into applications which have been exited.

T.Gen_Atk - Generational Attack

An attacker may use a TOE from another generation of issue to exploit differences in security function implementation to reveal security critical information.

3.2.1.6 Threats Regarding Cryptographic Functions

T.Crypt_Atk - Cryptographic Attack

An attacker may defeat security functions through a cryptographic attack against the algorithm or through a brute-force attack.

This attack may include either encode/decode functions or random number generators.

3.2.1.7 Threats which Monitor Information

T.IO_Man - Input/Output Manipulation

An attacker may manipulate connections to the IC and monitor the results to reveal critical security information.

Manipulation may involve direct control of the I/O, clock or power lines to generate security critical information either directly or through inference.

T.I_Leak - Information Leakage

An attacker may exploit information, which is leaked from the TOE during normal usage.

Leakage may occur through emanations, variations in power consumption, I/O characteristics, clock frequency, or by changes in processing time requirements.

T.Link - Linkage of Multiple Observations

An attacker may be able to observe multiple uses of resources or services by an entity and, by linking these uses, be able to deduce information which the entity wishes to be kept confidential.

3.2.1.8 Miscellaneous Threats

T.Lnk_Att - Linked Attacks

An attacker may perform successive attacks with the result that the TOE becomes unstable or some aspect of the security functionality is degraded. A following attack may then be successfully executed.

T.Env_Strs - Environmental Stress

An attacker may exploit failures in the TOE induced by environmental stress.

Exposure of the Integrated Circuit to conditions outside its specified operating range may result in malfunction or failure of security critical components allowing manipulation of programs or data.

3.2.2 Threats to the Environment

T.Dis_Des - Disclosure of Design

An attacker may gain access to the TOE through the unauthorized disclosure and use of hardware and software design information.

Design information may include: IC specification and technology, IC design, IC hardware security mechanisms, photomask, development tools, initialization procedures, access control mechanisms, authentication systems, data protection systems, memory partitioning, cryptographic programs, or pre-personalization requirements. Disclosure may occur either from within a specific process or in the transportation between elements in the card life-cycle.

T.Dis_Soft - Disclosure of Software

An attacker may gain access to the TOE through the unauthorized disclosure and use of TOE software.

This software may include IC software security mechanisms, initialization procedures, access control mechanisms, authentication systems, data protection systems, memory partitioning, or cryptographic programs. Disclosure may occur either from within a specific process or in the transportation between elements in the card life-cycle.

T.Dis_Data - Disclosure of Secret Data

An attacker may gain access to the TOE through the unauthorized disclosure and use of secret data.

This information may include initialization data, personalization data, passwords, or cryptographic keys. Disclosure may occur either from within a specific process or in the transportation between elements in the card life-cycle.

T.Dis_Test - Disclosure of Test Data

An attacker may gain access to the TOE through the unauthorized disclosure and use of TOE test data.

This information may include; test tools, test procedures, test programs, or test results. Disclosure may occur either from within a specific process or in the transportation between elements in the card life-cycle.

T.Tft_Prod - Theft of Product

An attacker may steal product for use in developing techniques and in compromising the security functions of the TOE.

Product which might be stolen can be silicon samples, bond-out chips, pre-initialized cards, pre-personalized cards, or personalized but unissued cards. Theft may occur either from within a specific process or in the transportation between elements in the card life-cycle.

T.Tft_Mask - Theft of Mask

An attacker may steal a TOE photomask to gain unauthorized understanding of the TOE leading to a compromise of the TOE security functions.

Theft may occur either from within a specific process or in the transportation between stages in the card life-cycle.

T.Tft_Tools - Theft of Tools

An attacker may steal TOE development tools to gain unauthorized understanding of the TOE leading to a compromise of the TOE security functions.

Theft may occur either from within a specific process or in the transportation between stages in the card life-cycle.

T.Mod_Des - Modification of Design

An attacker may modify TOE hardware and software design information to introduce flaws in security functionality which can be exploited later.

Information which, if modified, could lead to compromise may include: IC specification, IC hardware security mechanisms, photomask, development tools, initialization procedures, access control mechanisms, authentication systems, data protection systems, memory partitioning, or cryptographic programs. Modification of assets may occur either from within a specific process or in the transportation between elements in the card life-cycle.

T.Mod_Soft - Modification of Software

An attacker may modify TOE software to introduce flaws in security functionality which can be exploited later.

Software which, if modified, could lead to compromise may include IC software security mechanisms, initialization procedures, access control mechanisms, authentication systems, data protection systems, memory partitioning, or cryptographic programs. Modification of assets may occur either from within a specific process or in the transportation between elements in the card life-cycle.

T.Mod_Data - Modification of Secret Data

An attacker may modify TOE secret data to introduce flaws in security functionality which can be exploited later.

Secret data which, if modified, could lead to compromise may include initialization data, personalization data, passwords, or cryptographic keys. Modification of assets may occur either from within a specific process or in the transportation between elements in the card life-cycle.

T.Mod_Test - Modification of Test Data

An attacker may modify TOE test data to introduce flaws in security functionality which can be exploited later.

Test data which, if modified, could lead to may include; test tools, test procedures, test programs, or test results. Modification of assets may occur either from within a specific process or in the transportation between elements in the card life-cycle.

T.Key_Comp - Key Compromise

An attacker may gain access to the TOE through use of stolen or compromised cryptographic keys.

Key compromise may be through inadequate control processes or theft. It could occur at a particular site in the development and use of the card or could be exposed during transfer of key information between sites. Keys involved may be production keys, transport keys, test keys, or operational keys.

T.Clon – Cloning

An attacker may clone part or all of a functional TOE to develop further attacks.

3.3 Organizational Security Policies

The organizational security policies discussed below are addressed by VSCPP compliant TOEs.

P.Data_Acc - Data Access

Except for a well-defined set of allowed operations, the right to access specific data and data objects is determined on the basis of:

a) the owner of the object,

b) the identity of the subject attempting the access, and

c) the implicit and explicit access rights to the object granted to the subject by the object owner.

P.File_Acc - File Access

The right to establish files and the access control structure is determined on the basis of:

a) the owner of the files,

b) the identity of the subject attempting to perform setup, and

c) the implicit and explicit access rights to the files granted to the subject by the file's owner.

P.Mult_App - Multiple Applications

The interplay between core functions and applications, and particularly between multiple applications must conform to requirements determined by the owner of the core functions and the respective owners of the applications.

P.Crypt_Std - Cryptographic Standards

Cryptographic entities, data authentication, and approval functions must be in accordance with ISO and associated industry or organizational standards.

P.Ident - Identification

The TOE must be capable of being uniquely identified.

P.Sec_Com - Secure Communications

Secure communication protocols and procedures should be supported between the smartcard and terminal when required by the application.

P.IT_Std - Information Technology Standards

The TOE design should be in accordance with ISO and associated industry or organizational information technology standards.

P.Extend - Extension of Function

The TOE must be capable of modifying its operation after issue to allow additional, approved capabilities.

P.Con_Cont - Code Configuration Control

All code must be under configuration control.

4 Security Objectives

4.1 TOE Security Objectives

O.Phys_Prot - Physical Protection

The TOE must protect itself against physical compromise through having a structure which is resistant to physical attack or that creates difficulties in understanding information derived from such an attack

O.Log_Prot - Logical Protection

The TOE must protect itself against logical compromise through having a structure which is resistant to logical manipulation or modification.

O.DAC - Data Access Control

The TOE must provide its users with the means of controlling and limiting access to the objects and resources they own or are responsible for, on the basis of individual users or identified groups of users and in accordance with the set of rules defined by the P.Data_Acc Security Policy.

O.FAC - File Access Control

The TOE must provide its users with the means of controlling and limiting the ability to generate or modify files to the files and resources they own are responsible for, on the basis of individual users or identified groups of users and in accordance with the set of rules defined by the P.File_Acc Security Policy.

O.I_Leak - Information Leakage

The TOE must provide the means of controlling and limiting the leakage of information in the TOE such that no useful information is revealed over the power, ground, Clock, Reset, or I/O lines.

O.Set_Up - Set-Up Sequence

The TOE must require a defined sequence of operations prior to general utilization.

O. Mult_App - Multiple Applications

The TOE must support an application (or applications) while providing and maintaining security between and among the various resident elements.

O.Life_Cycle - Life Cycle Functions

The TOE must provide means of controlling and limiting the use of life cycle specific commands to the life cycle stages in which they are intended.

O.Crypt - Cryptography

The TOE must support cryptographic functions in a secure manner.

O.Search - Data Search

Data and files which are subject to search by unauthorized entities should be protected from repeated entry.

O.Flt_Ins - Fault Insertion

The TOE must be resistant to repeated probing through insertion of erroneous data.

O.Re-Use - Replay

Single use authentication shall be used for selected security functions to protect against replay attacks.

O.Ident - TOE Identification

The TOE must support the recording and preservation of identification information.

O.Init - Initialization

The TOE must assume its initial state immediately upon power-up, reset, or after other restart conditions.

O.D_Read - Data Read Format

The TOE must have a consistent requirement for formatting data passing between modules in the chip.

O.Sec_Com - Secure Communications

The TOE must be able to support secure communication protocols and procedures between the smartcard and terminal when required by the application.

O.Mem_Chk - Memory Integrity Checking

The TOE shall provide the means of detecting loss of integrity affecting security information stored in memories.

O.Extend - TOE Extensions

The TOE must, when properly specified and authorized, support modification or addition to its functionality.

O.Unlink - Linkage

The TOE must provide the means of allowing an entity to make multiple uses of resources or services without other entities being able to link those uses together.

O.Operate - Secure Operation

The TOE must ensure the continued correct operation of its security functions.

O.Flaw – Flaws

The TOE must not contain flaws in design, implementation or operation.

O.Admin - Administration

The TOE must provide functionality which enables an authorized administrator to effectively manage the TOE and its security functions, and must ensure that only authorized administrators are able to access such functionality.

O.IT_Std - IT Standards

The TOE must comply with relevant information technology processes and standards.

4.2 Environment Security Objectives

OE.Con_Des - Control of Design

Those responsible for the TOE must ensure that specifications, design information, details of hardware security mechanisms, IC specifications, IC databases, schematics/layout, software specification, detailed design, source code, or any further information are accessible only by authorized personnel.

OE.Con_Prod - Control of Product

The manufacturing process shall ensure the protection of the TOE from any kind of unauthorized use such as tampering or theft.

OE.Mask_Prot - Photomask Protection

The photomask fabrication management process shall ensure the protection of the mask from any kind of unauthorized use such as tampering or theft.

OE.Dlv_Proc - Delivery Procedures

Procedures shall ensure protection of TOE material/ information during delivery.

OE.Dlv_Aud - Delivery Audit

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

OE.Dlv_Trn - Delivery Training

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

OE.Sampl_Acs - Sample Access

Samples used to run test shall be accessible only by authorized personnel.

OE.Init_Acs - Initialization Access

Initialization Data shall be accessible only by authorized personnel.

OE.Perss - Personnel

Personnel working as administrators or in other privileged positions shall be carefully selected and trained for reliability.

OE.Key_Con - Crypto Key Control

All smart card related cryptographic keys must be controlled for confidentiality and integrity according to the owner's needs.

OE.Con_Cont - Code Configuration Control

All code must be under configuration control.

5 IT Security Requirements

5.1 TOE IT Security Requirements

This section contains the functional requirements that must be satisfied by a VSCPP compliant TOE. These requirements consist of functional components from Part 2 of the CC.

5.1.1 TOE IT Security Functional Requirements

Table 5.1 lists the IT security functional components and indicates whether the component has been refined and if all operations of that requirement are to be met by the TOE. Following the table, each requirement is listed with assignments, selections and refinements indicated in **bold** type. General assignments and selections, requiring definition in the ST are indicated in ***bold italic*** type.

Table 5.1 –Security Functional Components

Component	**Component Name**	**Refined?**	**Operations Done?**
FCS_CKM.3	Cryptographic key access	no	yes
FCS_COP.1	Cryptographic operation	no	yes
FDP_ACC.1	Subset access control	no	yes
FDP_ACF.1	Security attribute based access control	no	yes
FDP_ETC.1	Export of user data without security attributes	no	yes
FDP_IFC.1	Subset information flow control	no	yes
FDP_IFF.1	Simple security attributes	no	yes
FDP_ITC.1	Import of user data without security attributes	no	yes
FDP_ITT.1	Basic internal transfer protection	no	yes
FDP_RIP.2	Full residual information protection	no	yes
FDP_ROL.2	Advanced rollback	no	yes
FDP_SDI.2	Stored data integrity monitoring and action	no	yes
FDP_UIT.1	Data exchange integrity	no	yes

Component	Component Name	Refined?	Operations Done?
FIA_AFL.1	Authentication failure handling	no	yes
FIA_ATD.1	User attribute definition	no	yes
FIA_SOS.1	Verification of secrets	no	yes
FIA_SOS.2	TSF Generation of secrets	no	yes
FIA_UAU.1	Timing of authentication	no	yes
FIA_UAU.4	Single-use authentication mechanisms	no	yes
FIA_UAU.5	Multiple authentication mechanisms	no	yes
FIA_UAU.7	Protected authentication feedback	no	yes
FIA_UID.1	Timing of identification	no	yes
FMT_MOF.1	Management of security functions behavior	no	yes
FMT_MSA.1	Management of security attributes	no	yes
FMT_MSA.2	Secure security attributes	no	yes
FMT_MSA.3	Static attribute initialization	no	yes
FMT_MTD.1	Management of TSF data	no	yes
FMT_MTD.2	Management of limits on TSF data	no	yes
FMT_MTD.3	Secure TSF data	no	yes
FMT_REV.1	Revocation	no	yes
FPT_FLS.1	Failure with preservation of secure state	no	yes
FPT_ITI.1	Inter-TSF detection of modification	no	yes
FPT_ITT.1	Basic internal TSF data transfer protection	no	yes
FPT_PHP.3	Resistance to physical attack	no	yes
FPT_RCV.3	Automated recovery without undue loss	no	yes
FPT_RCV.4	Function recovery	no	yes
FPT_RPL.1	Replay detection	no	yes
FPT_RVM.1	Non-bypassability of the TSP	no	yes
FPT_SEP.1	TSF domain separation	no	yes
FPT_TST.1	TSF testing	yes	yes

FCS_CKM.3 - Cryptographic key access

FCS_CKM.3.1 The TSF shall perform ***type of cryptographic key access*** in accordance with a specified cryptographic key access method ***cryptographic key access method*** that meets the following***: list of standards*** .

FCS_COP.1 - Cryptographic operation

FCS_COP.1.1 The TSF shall perform ***list of cryptographic operations*** in accordance with a specified cryptographic algorithm ***cryptographic algorithm*** and cryptographic key *sizes* ***cryptographic key sizes*** that meet the following: ***list of standards*** .

FDP_ACC.1 - Subset access control

FDP-ACC.1.1 The TSF shall enforce the **Smart Card Access Control** on ***list of subjects, objects, and operations among subjects and objects covered by the SFP*** .

FDP_ACF.1 - Security attribute based access control

FDP-ACF.1.1 The TSF shall enforce the **Smart Card Access Control** to objects based on ***security attributes, named groups of 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: ***rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*** .

FDP-ACF.1.3 The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: ***rules, based on security attributes, that explicitly authorize access of subjects to objects***.

FDP-ACF.1.4 The TSF shall explicitly deny access of subjects to objects based on the ***rules, based on security attributes, that explicitly deny access of subjects to objects*** .

FDP_ETC.1 - Export of user data without security attributes

FDP-ETC.1.1 The TSF shall enforce the **Smart Card Access Control and Smart Card Information Flow Control** when exporting user data, controlled under the SFP(s), outside of the TSC.

FDP-ETC.1.2 The TSF shall export the user data without the user data's associated security attributes.

FDP_IFC.1 - Subset information flow control

FDP-IFC.1.1 The TSF shall enforce the **Smart Card Information Flow Control** on ***list of subjects, information, and operations that cause controlled information to flow to and from controlled subjects covered by the SFP*** .

FDP_IFF.1 - Simple security attributes

FDP-IFF.1.1 The TSF shall enforce the **Smart Card Information Flow Control** based on the following types of subject and information security attributes: ***the minimum number and type of security attributes*** .

FDP-IFF.1.2 The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: ***for each operation, the security attribute-based relationship that must hold between subject and information security attributes*** .

FDP-IFF.1.3 The TSF shall enforce the ***additional information flow control SFP rules*** .

FDP-IFF.1.4 The TSF shall provide the following ***list of additional SFP capabilities*** .

FDP-IFF.1.5 The TSF shall explicitly authorize an information flow based on the following rules*:* ***rules, based on security attributes, that explicitly authorize information flows*** .

FDP-IFF.1.6 The TSF shall explicitly deny an information flow based on the following rules*:* ***rules, based on security attributes, that explicitly deny information flows*** .

FDP_ITC.1 - Import of user data without security attributes

FDP-ITC.1.1 The TSF shall enforce the **Smart Card Access Control and Smart Card Information Flow Control** when importing user data, controlled under the SFP, from outside of the TSC.

FDP-ITC.1.2 The TSF shall ignore any security attributes associated with the user data when imported from outside the TSC.

FDP-ITC.1.3 The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TSC: ***additional importation control rules*** .

FDP_ITT.1 - Basic internal transfer protection

FDP-ITT.1.1 The TSF shall enforce the **Smart Card Access Control and Smart Card Information Flow Control** to prevent the **disclosure or modification** of user data when it is transmitted between physically-separated parts of the TOE.

FDP_RIP.2 - Full residual information protection

FDP-RIP.2.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the **de-allocation of the resource from** all objects.

FDP_ROL.2 - Advanced rollback

FDP-ROL.2.1 The TSF shall enforce **Smart Card Information Flow Control** to permit the rollback of all the operations on the ***list of objects*** .

FDP-ROL.2.2 The TSF shall permit operations to be rolled back within the **boundary limit of the task being performed when operation is prematurely terminated** .

FDP_SDI.2 - Stored data integrity monitoring and action

FDP-SDI.2.1 The TSF shall monitor user data stored within the TSC for ***integrity errors*** on all objects, based on the following attributes: ***user data attributes*** .

FDP-SDI.2.2 Upon detection of a data integrity error, the TSF shall ***action to be taken*** .

FDP_UIT.1 - Data exchange integrity

FDP-UIT.1.1 The TSF shall enforce the **Smart Card Information Flow Control** to be able to **transmit** user data in a manner protected from **modification** errors.

FDP-UIT.1.2 The TSF shall be able to determine on receipt of user data, whether **modification** has occurred.

FIA_AFL.1 - Authentication failure handling

FIA_AFL.1.1 The TSF shall detect when ***number*** unsuccessful authentication attempts occur related to ***list of authentication events*** .

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF shall ***list of actions***.

FIA_ATD.1 - User attribute definition

FIA_ATD.1.1 The TSF shall maintain the following list of security attributes belonging to individual users: ***list of security attributes*** .

FIA_SOS.1 - Verification of secrets

FIA_SOS.1.1 The TSF shall provide a mechanism to verify that secrets meet ***a defined quality metric***

FIA_SOS.2 - TSF Generation of secrets

FIA_SOS.2.1 he TSF shall provide a mechanism to generate secrets that meet ***a defined quality metric*** .

FIA_SOS.2.2 The TSF shall be able to enforce the use of TSF generated secrets for ***list of TSF functions*** .

FIA_UAU.1 - Timing of authentication

FIA_UAU.1.1 The TSF shall allow ***list of TSF mediated actions*** on behalf of the user to be performed before the user is authenticated.

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

FIA_UAU.4 - Single-use authentication mechanisms

FIA_UAU.4.1 The TSF shall prevent reuse of authentication data related to ***identified authentication mechanism(s)***.

FIA_UAU.5 - Multiple authentication mechanisms

FIA_UAU.5.1 The TSF shall provide ***list of multiple authentication mechanisms*** to support user authentication.

FIA_UAU.5.2 The TSF shall authenticate any user’s claimed identity according to the ***rules describing how the multiple authentication mechanisms provide authentication***.

FIA_UAU.7 - Protected authentication feedback

FIA_UAU.7.1 The TSF shall provide only **none** to the user while the authentication is in progress.

FIA_UID.1 - Timing of identification

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

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

FMT_MOF.1 - Management of security functions behavior

FMT_MOF.1.1 The TSF shall restrict the ability to ***determine the behavior of, disable, enable, modify the behavior of*** the functions ***list of functions*** to ***the authorized identified roles***.

FMT_MSA.1 - Management of security attributes

FMT_MSA.1.1 The TSF shall enforce the **Smart Card Access Control, Smart Card Information Control** to restrict the ability to ***change_default, query, modify, delete, other operations*** the security attributes ***list of security attributes*** to ***the authorized identified roles***.

FMT_MSA.2 - Secure security attributes

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

FMT_MSA.3 - Static attribute initialization

FMT_MSA.3.1 The TSF shall enforce the **Smart Card Access Control, Smart Card Information Flow Control** to provide **restrictive** default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 The TSF shall allow the ***authorized identified roles*** to specify alternative initial values to override the default values when an object or information is created.

FMT_MTD.1 - Management of TSF data

FMT_MTD.1.1 The TSF shall restrict the ability to ***change_default, query, modify, delete, clear, other operations*** the ***list of TSF data*** to ***the authorized identified roles***.

FMT_MTD.2 - Management of limits on TSF data

FMT_MTD.2.1 The TSF shall restrict the specification of the limits for ***list of TSF data*** to ***the authorized identified roles***.

FMT_MTD.2.2 The TSF shall take the following actions, if the TSF data are at, or exceed, the indicated limits: ***actions to be taken***.

FMT_MTD.3 - Secure TSF data

FMT_MTD.3.1 The TSF shall ensure that only secure values are accepted for TSF data.

FMT_REV.1 - Revocation

FMT_REV.1.1 The TSF shall restrict the ability to revoke security attributes associated with the ***users, subjects, objects, other additional resources*** within the TSC to ***the authorized identified roles***.

FMT_REV.1.2 The TSF shall enforce the rules ***specification of revocation rules*** .

FPT_FLS.1 - Failure with preservation of secure state

FPT_FLS.1.1 The TSF shall preserve a secure state when the following types of failures occur: ***list of types of failures in the TSF***.

FPT_ITI.1 - Inter-TSF detection of modification

FPT_ITI.1.1 The TSF shall provide the capability to detect modification of all TSF data during transmission between the TSF and a remote trusted IT product within the following metric: ***a defined modification metric***.

FPT_ITI.1.2 The TSF shall provide the capability to verify the integrity of all TSF data transmitted between the TSF and a remote trusted IT product and perform ***action to be taken*** if modifications are detected.

FPT_ITT.1 - Basic internal TSF data transfer protection

FPT_ITT.1.1 The TSF shall protect TSF data from **modification** when it is transmitted between separate parts of the TOE.

FPT_PHP.3 - Resistance to physical attack

FPT_PHP.3.1 The TSF shall resist **environmental stress** to the ***list of TSF devices/elements*** by responding automatically such that the TSP is not violated.

FPT_RCV.3 - Automated recovery without undue loss

FPT_RCV.3.1 When automated recovery from a failure or service discontinuity is not possible, the TSF shall enter a maintenance mode where the ability to return the TOE to a secure state is provided.

FPT_RCV.3.2 For **power failure during operation** , the TSF shall ensure the return of the TOE to a secure state using automated procedures.

FPT_RCV.3.3 The functions provided by the TSF to recover from failure or service discontinuity shall ensure that the secure initial state is restored without exceeding ***quantification*** for loss of TSF data or objects within the TSC.

FPT_RCV.3.4 The TSF shall provide the capability to determine the objects that were or were not capable of being recovered.

FPT_RCV.4 - Function recovery

FPT_RCV.4.1 The TSF shall ensure that **for the security functions involved in rollback and reset functions and the scenario of power loss or smart card withdrawal prior to completion** have the property that the SF either completes successfully, or for the indicated failure scenarios, recovers to a consistent and secure state.

FPT_RPL.1 - Replay detection

FPT_RPL.1.1 The TSF shall detect replay for the following entities: ***list of identified entities***.

FPT_RPL.1.2 The TSF shall perform ***list of specific actions*** when replay is detected.

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.

FPT_SEP.1 - TSF domain separation

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

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

FPT_TST.1 - TSF testing

FPT_TST.1.1 The TSF shall run a suite of self tests **during initial start-up and at the conditions under which self test should occur** to demonstrate the correct operation of the TSF.

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

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

Refined by adding:

Self testing should include the functions providing the card blocking function and other functions as detailed in the Functional Specification.

5.1.2 TOE IT Security Assurance Requirements

Table 5.2 lists the IT security assurance components and indicates whether the component has been refined. Following the table, each requirement is listed with refinements identified. These requirements are chosen to be consistent with an EAL4 augmented assurance level. Augmentation includes AVA_VLA.3 and ADV_INT.1.

Table 5.2 – Security Assurance Components

Component	**Component Name**	**Refined?**
ACM_AUT.1	Partial CM automation	no
ACM_CAP.4	Generation support and acceptance procedures	no
ACM_SCP.2	Problem tracking CM coverage	no
ADO_DEL.2	Detection of modification	no
ADO_IGS.1	Installation, generation, and start-up procedures	no
ADV_FSP.2	Fully defined external interfaces	no
ADV_HLD.2	Security enforcing high-level design	no
ADV_IMP.1	Subset of the implementation of the TSF	yes
ADV_INT.1	Modularity	no
ADV_LLD.1	Descriptive low-level design	no
ADV_RCR.1	Informal correspondence demonstration	no
ADV_SPM.1	Informal TOE security policy model	no
AGD_ADM.1	Administrator guidance	no
AGD_USR.1	User guidance	no
ALC_DVS.1	Identification of security measures	no
ALC_LCD.1	Developer defined life-cycle model	no
ALC_TAT.1	Well-defined development tools	no
ATE_COV.2	Analysis of coverage	no
ATE_DPT.1	Testing: high-level design	no
ATE_FUN.1	Functional testing	no

Component	Component Name	Refined?
ATE_IND.2	Independent testing - sample	no
AVA_MSU.2	Validation of analysis	no
AVA_SOF.1	Strength of TOE security function evaluation	no
AVA_VLA.3	Moderately resistant	yes

ACM_AUT.1 - Partial CM automation

Developer action elements:

ACM_AUT.1.1D The developer shall use a CM system.

ACM_AUT.1.2D The developer shall provide a CM plan.

Content and presentation of evidence elements:

ACM_AUT.1.1C The CM system shall provide an automated means by which only authorized changes are made to the TOE implementation representation.

ACM_AUT.1.2C The CM system shall provide an automated means to support the generation of the TOE.

ACM_AUT.1.3C The CM plan shall describe the automated tools used in the CM system.

ACM_AUT.1.4C The CM plan shall describe how the automated tools are used in the CM system.

Evaluator action elements:

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

ACM_CAP.4 - Generation support and acceptance procedures

Developer action elements:

ACM_CAP.4.1D The developer shall provide a reference for the TOE.

ACM_CAP.4.2D The developer shall use a CM system.

ACM_CAP.4.3D The developer shall provide CM documentation.

Content and presentation of evidence elements:

ACM_CAP.4.1C The reference for the TOE shall be unique to each version of the TOE.

ACM_CAP.4.2C The TOE shall be labeled with its reference.

ACM_CAP.4.3C The CM documentation shall include a configuration list, a CM plan, and an acceptance plan.

ACM_CAP.4.4C The configuration list shall describe the configuration items that comprise the TOE.

ACM_CAP.4.5C The CM documentation shall describe the method used to uniquely identify the configuration items.

ACM_CAP.4.6C The CM system shall uniquely identify all configuration items.

ACM_CAP.4.7C The CM plan shall describe how the CM system is used.

ACM_CAP.4.8C The evidence shall demonstrate that the CM system is operating in accordance with the CM plan.

ACM_CAP.4.9C The CM documentation shall provide evidence that all configuration items have been and are being effectively maintained under the CM system.

ACM_CAP.4.10C The CM system shall provide measures such that only authorized changes are made to the configuration items.

ACM_CAP.4.11C The CM system shall support the generation of the TOE.

ACM_CAP.4.12C The acceptance plan shall describe the procedures used to accept modified or newly created configuration items as part of the TOE.

Evaluator action elements:

ACM_CAP.4.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ACM_SCP.2 - Problem tracking CM coverage

Developer action elements:

ACM_SCP.2.1D The developer shall provide CM documentation.

Content and presentation of evidence elements:

ACM_SCP.2.1C The CM documentation shall show that the CM system, as a minimum, tracks the following: the TOE implementation representation, design documentation, test documentation, user documentation, administrator documentation, CM documentation, and security flaws.

ACM_SCP.2.2C The CM documentation shall describe how configuration items are tracked by the CM system.

Evaluator action elements:

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

ADO_DEL.2 - Detection of modification

Developer action elements:

ADO_DEL.2.1D The developer shall document procedures for delivery of the TOE or parts of it to the user.

ADO_DEL.2.2D The developer shall use the delivery procedures.

Content and presentation of evidence elements:

ADO_DEL.2.1C The delivery documentation shall describe all procedures that are necessary to maintain security when distributing versions of the TOE to a user׳s site.

ADO_DEL.2.2C The delivery documentation shall describe how the various procedures and technical measures provide for the detection of modifications, or any discrepancy between the developer׳s master copy and the version received at the user site.

ADO_DEL.2.3C The delivery documentation shall describe how the various procedures allow detection of attempts to masquerade as the developer, even in cases in which the developer has sent nothing to the user׳s site.

Evaluator action elements:

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

ADO_IGS.1 - Installation, generation, and start-up proce dures

Developer action elements:

ADO_IGS.1.1D The developer shall document procedures necessary for the secure installation, generation, and start-up of the TOE.

Content and presentation of evidence elements:

ADO_IGS.1.1C The documentation shall describe the steps necessary for secure installation, generation, and start-up of the TOE.

Evaluator action elements:

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

ADO_IGS.1.2E The evaluator shall determine that the installation, generation, and start-up procedures result in a secure configuration.

ADV_FSP.2 - Fully defined external interfaces

Dependencies:

ADV_RCR.1 Informal correspondence demonstration

Developer action elements:

ADV_FSP.2.1D The developer shall provide a functional specification.

Content and presentation of evidence elements:

ADV_FSP.2.1C The functional specification shall describe the TSF and its external interfaces using an informal style.

ADV_FSP.2.2C The functional specification shall be internally consistent.

ADV_FSP.2.3C The functional specification shall describe the purpose and method of use of all external TSF interfaces, providing complete details of all effects, exceptions and error messages.

ADV_FSP.2.4C The functional specification shall completely represent the TSF.

ADV_FSP.2.5C The functional specification shall include rationale that the TSF is completely represented.

Evaluator action elements:

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

ADV_FSP.2.2E The evaluator shall determine that the functional specification is an accurate and complete instantiation of the TOE security functional requirements.

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.

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 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.

ADV_IMP.1 - Subset of the implementation of the TSF

Developer action elements:

ADV_IMP.1.1D The developer shall provide the implementation representation for a selected subset of the TSF.

Content and presentation of evidence elements:

ADV_IMP.1.1C The implementation representation shall unambiguously define the TSF to a level of detail such that the TSF can be generated without further design decisions.

ADV_IMP.1.2C The implementation representation shall be internally consistent.

Evaluator action elements:

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

ADV_IMP.1.2E The evaluator shall determine that the least abstract TSF representation provided is an accurate and complete instantiation of the TOE security functional requirements.

Refined by adding:

The selected subsets for evaluation shall include:

the subset of the physical structure of the TOE related to:

- structure size, organization, and layout
- interconnects and data bus layout
- fuse locations
- physical structure including shielding layers and packaging.
- EEPROM manipulation
- RAM access
- command range and validity checking
- secret data checking and manipulation
- availability of commands outside of defined application
- transfer of information between applications or functions.

the subset of the structure of the TOE providing unalterability of selected data including:

- serial number and other life-cycle identifiers
- blocking or elimination of debugging functions
- first time use indicator
- configuration of blocking functions
- card disablement indicator.

the subset of the structure of the TOE providing the interrupts and reset function.

ADV_INT.1 - Modularity

Developer action elements:

ADV_INT.1.1D The developer shall design and structure the TSF in a modular fashion that avoids unnecessary interactions between the modules of the design.

ADV_INT.1.2D The developer shall provide an architectural description.

Content and presentation of evidence elements:

ADV_INT.1.1C The architectural description shall identify the modules of the TSF.

ADV_INT.1.2C The architectural description shall describe the purpose, interface, parameters, and effects of each module of the TSF.

ADV_INT.1.3C The architectural description shall describe how the TSF design provides for largely independent modules that avoid unnecessary interactions.

Evaluator action elements:

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

ADV_INT.1.2E The evaluator shall determine that both the low-level design and the implementation representation are in compliance with the architectural description.

ADV_LLD.1 - Descriptive low-level design

Developer action elements:

ADV_LLD.1.1D The developer shall provide the low-level design of the TSF.

Content and presentation of evidence elements:

ADV_LLD.1.1C The presentation of the low-level design shall be informal.

ADV_LLD.1.2C The low-level design shall be internally consistent.

ADV_LLD.1.3C The low-level design shall describe the TSF in terms of modules.

ADV_LLD.1.4C The low-level design shall describe the purpose of each module.

ADV_LLD.1.5C The low-level design shall define the interrelationships between the modules in terms of provided security functionality and dependencies on other modules.

ADV_LLD.1.6C The low-level design shall describe how each TSP-enforcing function is provided.

ADV_LLD.1.7C The low-level design shall identify all interfaces to the modules of the TSF.

ADV_LLD.1.8C The low-level design shall identify which of the interfaces to the modules of the TSF are externally visible.

ADV_LLD.1.9C The low-level design shall describe the purpose and method of use of all interfaces to the modules of the TSF, providing details of effects, exceptions and error messages, as appropriate.

ADV_LLD.1.10C The low-level design shall describe the separation of the TOE into TSP-enforcing and other modules.

Evaluator action elements:

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

ADV_LLD.1.2E The evaluator shall determine that the low-level design is an accurate and complete instantiation of the TOE security functional requirements.

ADV_RCR.1 - Informal correspondence demonstration

Developer action elements:

ADV_RCR.1.1D The developer shall provide an analysis of correspondence between all adjacent pairs of TSF representations that are provided.

Content and presentation of evidence elements:

ADV_RCR.1.1C For each adjacent pair of provided TSF representations, the analysis shall demonstrate that all relevant security functionality of the more abstract TSF representation is correctly and completely refined in the less abstract TSF representation.

Evaluator action elements:

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

ADV_SPM.1 - Informal TOE security policy model

Developer action elements:

ADV_SPM.1.1D The developer shall provide a TSP model.

ADV_SPM.1.2D The developer shall demonstrate correspondence between the functional specification and the TSP model.

Content and presentation of evidence elements:

ADV_SPM.1.1C The TSP model shall be informal.

ADV_SPM.1.2C The TSP model shall describe the rules and characteristics of all policies of the TSP that can be modeled.

ADV_SPM.1.3C The TSP model shall include a rationale that demonstrates that it is consistent and complete with respect to all policies of the TSP that can be modeled.

ADV_SPM.1.4C The demonstration of correspondence between the TSP model and the functional specification shall show that all of the security functions in the functional specification are consistent and complete with respect to the TSP model.

Evaluator action elements:

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

AGD_ADM.1 - Administrator guidance

Developer action elements:

AGD_ADM.1.1D The developer shall provide administrator guidance addressed to system administrative personnel.

Content and presentation of evidence elements:

AGD_ADM.1.1C The administrator guidance shall describe the administrative functions and interfaces available to the administrator of the TOE.

AGD_ADM.1.2C The administrator guidance shall describe how to administer the TOE in a secure manner.

AGD_ADM.1.3C The administrator guidance shall contain warnings about functions and privileges that should be controlled in a secure processing environment.

AGD_ADM.1.4C The administrator guidance shall describe all assumptions regarding user behavior that are relevant to secure operation of the TOE.

AGD_ADM.1.5C The administrator guidance shall describe all security parameters under the control of the administrator, indicating secure values as appropriate.

AGD_ADM.1.6C The administrator guidance shall describe each type of security-relevant event relative to the administrative functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.

AGD_ADM.1.7C The administrator guidance shall be consistent with all other documentation supplied for evaluation.

AGD_ADM.1.8C The administrator guidance shall describe all security requirements for the IT environment that are relevant to the administrator.

Evaluator action elements:

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

AGD_USR.1 - User guidance

Developer action elements:

AGD_USR.1.1D The developer shall provide user guidance.

Content and presentation of evidence elements:

AGD_USR.1.1C The user guidance shall describe the functions and interfaces available to the non-administrative users of the TOE.

AGD_USR.1.2C The user guidance shall describe the use of user-accessible security functions provided by the TOE.

AGD_USR.1.3C The user guidance shall contain warnings about user-accessible functions and privileges that should be controlled in a secure processing environment.

AGD_USR.1.4C The user guidance shall clearly present all user responsibilities necessary for secure operation of the TOE, including those related to assumptions regarding user behavior found in the statement of TOE security environment.

AGD_USR.1.5C The user guidance shall be consistent with all other documentation supplied for evaluation.

AGD_USR.1.6C The user guidance shall describe all security requirements for the IT environment that are relevant to the user.

Evaluator action elements:

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

ALC_DVS.1 - Identification of security measures

Developer action elements:

ALC_DVS.1.1D The developer shall produce development security documentation.

Content and presentation of evidence elements:

ALC_DVS.1.1C The development security documentation shall describe all the physical, procedural, personnel, and other security measures that are necessary to protect the confidentiality and integrity of the TOE design and implementation in its development environment.

ALC_DVS.1.2C The development security documentation shall provide evidence that these security measures are followed during the development and maintenance of the TOE.

Evaluator action elements:

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

ALC_DVS.1.2E The evaluator shall confirm that the security measures are being applied.

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.

Evaluator action elements:

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

ALC_TAT.1 - Well-defined development tools

Developer action elements:

ALC_TAT.1.1D The developer shall identify the development tools being used for the TOE.

ALC_TAT.1.2D The developer shall document the selected implementation-dependent options of the development tools.

Content and presentation of evidence elements:

ALC_TAT.1.1C All development tools used for implementation shall be well-defined.

ALC_TAT.1.2C The documentation of the development tools shall unambiguously define the meaning of all statements used in the implementation.

ALC_TAT.1.3C The documentation of the development tools shall unambiguously define the meaning of all implementation-dependent options.

Evaluator action elements:

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

ATE_COV.2 - Analysis of coverage

Developer action elements:

ATE_COV.2.1D The developer shall provide an analysis of the test coverage.

Content and presentation of evidence elements:

ATE_COV.2.1C The analysis of the test coverage shall demonstrate the correspondence between the tests identified in the test documentation and the TSF as described in the functional specification.

ATE_COV.2.2C The analysis of the test coverage shall demonstrate that the correspondence between the TSF as described in the functional specification and the tests identified in the test documentation is complete.

Evaluator action elements:

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

ATE_DPT.1 - Testing: high-level design

Developer action elements:

ATE_DPT.1.1D The developer shall provide the analysis of the depth of testing.

Content and presentation of evidence elements:

ATE_DPT.1.1C The depth analysis shall demonstrate that the tests identified in the test documentation are sufficient to demonstrate that the TSF operates in accordance with its high-level design.

Evaluator action elements:

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

ATE_FUN.1 - Functional testing

Dependencies: **No dependencies.**

Developer action elements:

ATE_FUN.1.1D The developer shall test the TSF and document the results.

ATE_FUN.1.2D The developer shall provide test documentation.

Content and presentation of evidence elements:

ATE_FUN.1.1C The test documentation shall consist of test plans, test procedure descriptions, expected test results and actual test results.

ATE_FUN.1.2C The test plans shall identify the security functions to be tested and describe the goal of the tests to be performed.

ATE_FUN.1.3C The test procedure descriptions shall identify the tests to be performed and describe the scenarios for testing each security function. These scenarios shall include any ordering dependencies on the results of other tests.

ATE_FUN.1.4C The expected test results shall show the anticipated outputs from a successful execution of the tests.

ATE_FUN.1.5C The test results from the developer execution of the tests shall demonstrate that each tested security function behaved as specified.

Evaluator action elements:

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

ATE_IND.2 - Independent testing –sample

Developer action elements:

ATE_IND.2.1D The developer shall provide the TOE for testing.

Content and presentation of evidence elements:

ATE_IND.2.1C The TOE shall be suitable for testing.

ATE_IND.2.2C The developer shall provide an equivalent set of resources to those that were used in the developer’s functional testing of the TSF.

Evaluator action elements:

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

AVA_MSU.2 - Validation of analysis

Developer action elements:

AVA_MSU.2.1D The developer shall provide guidance documentation.

AVA_MSU.2.2D The developer shall document an analysis of the guidance documentation.

Content and presentation of evidence elements:

AVA_MSU.2.1C The guidance documentation shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation.

AVA_MSU.2.2C The guidance documentation shall be complete, clear, consistent and reasonable.

AVA_MSU.2.3C The guidance documentation shall list all assumptions about the intended environment.

AVA_MSU.2.4C The guidance documentation shall list all requirements for external security measures (including external procedural, physical and personnel controls).

AVA_MSU.2.5C The analysis documentation shall demonstrate that the guidance documentation is complete.

Evaluator action elements:

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

AVA_MSU.2.2E The evaluator shall repeat all configuration and installation procedures, and other procedures selectively, to confirm that the TOE can be configured and used securely using only the supplied guidance documentation.

AVA_MSU.2.3E The evaluator shall determine that the use of the guidance documentation allows all insecure states to be detected.

AVA_MSU.2.4E The evaluator shall confirm that the analysis documentation shows that guidance is provided for secure operation in all modes of operation of the TOE.

AVA_SOF.1 - Strength of TOE security function evaluation

Developer action elements:

AVA_SOF.1.1D The developer shall perform a strength of TOE security function analysis for each mechanism identified in the ST as having a strength of TOE security function claim.

Content and presentation of evidence elements:

AVA_SOF.1.1C For each mechanism with a strength of TOE security function claim the strength of TOE security function analysis shall show that it meets or exceeds the minimum strength level defined in the PP/ST.

AVA_SOF.1.2C For each mechanism with a specific strength of TOE security function claim the strength of TOE security function analysis shall show that it meets or exceeds the specific strength of function metric defined in the PP/ST.

Evaluator action elements:

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

AVA_SOF.1.2E The evaluator shall confirm that the strength claims are correct.

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.

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.

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.

Refined by adding:

The vulnerabilities to be analyzed shall include at least the following:

V_1 Access may be gained to a protected area of the TOE through connecting an externally “open”area to the protected area through a conductive “bridge.”

V_2 Modification to programmable nonvolatile memory code can compromise security.

V_3 Use of invalid or out of range commands and addresses may open the TOE to logical probing.

V_4 Cross application references may open the TOE to logical probing.

V_5 Use of native COS commands which are not required for the execution of an application might compromise the security of that application.

V_6 Attempting to by-pass a blown fuse within the TOE may allow otherwise precluded operations.

V_7 Data buses and conductive lines inside the TOE may be probed for information.

V_8 Out of environment conditions may stress the TOE such that proper operation is compromised.

V_9 Invocation of the debug mode may allow improper access to TOE functions.

V_10 Tapping the interconnects among functional blocks of the TOE may provide unauthorized access to information.

V_11 Electronically reading RAM data inside the TOE may provide unauthorized access to that information.

V_12 Invocation of TOE specific life cycle commands outside of the life cycle to which they are specific may allow improper access to TOE functions.

V_13 Modification or deletion of identification information in the TOE may allow unauthorized use of the TOE.

V_14 Preempting the COS through external commands, signals, or timed resets may allow improper access to TOE functions.

V_15 Power cut or transaction interruption may force the TOE into an insecure state.

V_16 Removal of material to expose the underlying structure may allow physical probing of the TOE potentially revealing design information and operational contents.

V_17 Leakage of information through emanations, variations in power consumption, I/O characteristics, clock frequency or processing time may reveal TOE operational contents.

5.2 Security Requirements for the IT Environment

Table 5.3 lists the IT security assurance components that apply to the IT Environment. No refinements are required.

Table 5.3 – Security Requirements for the Environment

Component	Component Name	Refined?
FCS_CKM.1	Cryptographic key generation	no
FCS_CKM.4	Cryptographic key destruction	no
FMT_SMR.1	Security Roles	no

FCS_CKM.1 - Cryptographic key generation

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified ***cryptographic key generation algorithm*** and specified ***cryptographic key sizes*** that meet the following ***list of standards*****.**

FCS_CKM.4 - Cryptographic key destruction

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified ***cryptographic key destruction method*** that meets the following ***list of standards*****.**

FMT_SMR.1 - Security roles

FMT_SMR.1.1 The TSF shall maintain ***the authorized identified roles*****.**

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

6 PP Application Notes

6.1 Issues Unique to Smart Cards

The Common Criteria was written against a background of traditional Information Technology, which generally discusses networked computers. This is very clear in considering the components in the Security Functional Requirements section of the Common Criteria, and indeed, throughout. Smart Cards have unique features that ripple through the entire process. The primary features that impact on security and the Common Criteria are discussed in this section.

6.1.1 Cost and Availability of the TOE

Smart cards range in cost from a few dollars to about twenty dollars. Most of the products envisaged by the Common Criteria (software for networked computers) cost tens of thousands of dollars per copy. This means that attackers can be expected to be able to buy multiple copies of the smart card TOE to experiment with, and destroying some of them in the course of exploration may be considered normal practice. That is not simply a matter of listing a new threat; it requires rethinking all threats in terms of probability and ease.

Most successful smart card projects anticipate issuing hundreds of thousands if not millions of the same card. This has critically important security implications.

- Attackers should be assumed to be able to get multiple copies of cards, unlike software products for networked computers.
- The asset protected by a single card may be low in value, but the total assets protected by the total card base may be very large.
- The cost of attacking a single card may not be worth the effort, but if that successful attack makes subsequent attacks on similar cards easy, the aggregate benefit may justify the effort. Initial attacks may require expensive reverse engineering of the smart card, after which subsequent attacks may be much easier and faster. Evaluation methodology must include estimates of the cost and difficulty of subsequent attacks as well as the initial attack.

6.1.2 Possession

Smart cards are in the possession of the cardholder all the time. The cardholder may be motivated to fraudulently change some of the data on the card (e.g., balance on a stored value card, age on an identification card, etc.). An attacker may be attacking his own card, or may steal one or several of them. The attacker can take it to a well equipped lab and subject it to all sorts of attacks. With the usual type of networked computer software product, this isnt possible.

Any estimate of the time it takes to conduct an attack must factor in the fact that the attacker has complete control of the card. It may not matter that it takes months to succeed, if the reward is high enough. There is sometimes an assumption that if an attack takes a long time, there will be adequate opportunity for detection of the attack. That does not apply when the attacker has complete physical control of the card which is not connected to anything that might detect an assault.

6.1.3 Roles

The Common Criteria discusses the roles of Developer, Administrator, and User, assuming that these are an exhaustive list. In some applications, there may be additional roles that are Administrative in some senses and Users in others. Examples are bank/Issuer clerk, merchant clerk, doctor, nurse, pharmacist, etc. Each application must specify its roles and their attendant privileges and map these to the appropriate Common Criteria components.

6.1.4 Off-line Operation

Most of the products that the Common Criteria was designed to evaluate are constantly on a network when in use. Smart cards are often used off-line; that is one of their significant advantages. Any counter-measures that depend on real time network monitoring will be ineffective in such instances.

6.1.5 Limited Memory & Processing Power

Most smart cards in 1999 use 8 bit microprocessors. Although more powerful 16 and even 32 bit chips will be available shortly, none have multi-threading and other powerful features that are common in standard computers.

Memory sizes range from as little as 1K of programmable nonvolatile memory to as much as 24K, with larger memory chips coming soon. ROM size is similarly limited. However large they become, they will always be relatively limited. That requires unique discipline in coding and limits the defensive measures that can be implemented.

6.1.6 Cost Sensitivity

The markets for smart cards are highly cost sensitive; differences of a few cents per card matter when millions of units are involved. That means that any defensive measures must meet very stringent cost effectiveness tests that are unusual with other IT products.

6.1.7 Physical Attacks

Generally, it is possible to evaluate logical attacks separately from physical attacks. This can be done with smart cards to some extent, but not entirely. Physical attacks utilizing techniques derived from semiconductor engineering must be evaluated or the whole effort is inadequate. The fact that the smart card gets its power and clock signal from outside the card imposes unique requirements, vulnerabilities, and protections. Just as there is a unique synergy in the way that a smart card uses hardware and software together to accomplish its tasks, attacks can also use a combination of hardware and software. Hardware-based defenses that might be effective can be breached by software that does not know how to use those defenses to best advantage.

Evaluation procedures and facilities must bring to bear expertise in hardware engineering as well as the more familiar software engineering and cryptography.

6.2 Smart Card Security Function Policies

The discussions above identify a number of issues which require detailed requirements beyond the simple statement of the need for a policy to be supplied by the ST. The policies for access control and information flow control have such a need for specificity, identifying the basic initial requirements which must be met by VSCPP compliant smart cards while leaving the freedom to add or modify applications having impact on the implementation of functions.

6.2.1 Access Control Security Function Policy

Component:FDP_ACC.1

Subset access control, identifies the need for an access control security function policy. This access control SFP is also referred to in other components.

The access control SFP for compliance with VSCPP includes the following elements:

Roles File and data access rights will be defined and only certain roles will be granted access privileges.

Memory Access to memory shall be solely controlled through the COS.

DataLoad All loading of data into the TOE requires authorization.

Access Levels Access conditions, once set, shall apply to all access and shall never be downgraded.

File Control The process and commands for creating the application file structure, including file access conditions, shall be controlled by access control provisions which are used only for this purpose. The file structure for an application, once created, may be locked from any future modification or deletion.

Crypt1 The PIN and other secret data, including cryptographic data, must be stored using access control provisions such that they cannot be read from outside.

Crypt2 The COS must provide a way to separate cryptographic environments, applications, and keys from each other during differing applications or stages in the life cycle.

Alternate Functions Control mechanisms for sensitive items shall not be used during routine operations or exposed by any other functions. If other core functions are used by non-COS applications, the applications must utilize the functions and process security controls equal to that in the core functions.

First-Use Authentication should be performed on first use (when applicable). Indication of first use shall not be alterable.

Card Block Blocking of the card must prevent access to all functions by the cardholder and any entity other than that defined by the operating system.

Card Disable Disabling the TOE shall prevent any further use and shall be non-reversible.

Application Application dependent access control provisions shall be fully described.

Other Additional access control SFP elements shall be specified as appropriate.

6.2.2 Information Flow Control Security Function Policy

Component:FDP_IFC.1

Subset information flow control, identifies the need for an information flow control security function policy. This information flow control SFP is also referred to in other components.

The information flow control SFP for compliance with VSCPP includes the following elements:

Information-Flow Data which is passing between modules (physical or functional areas) in the chip must be transmitted in the format in which the data will be stored

Identifiers Whenever possible, life cycle identifiers should be preserved and readable on a disabled card.

Alternate Function Control mechanisms for sensitive items shall not be used during routine operations or exposed by any other functions. If other core functions are used by non-COS applications, the applications must utilize the functions and process security controls equal to that in the core functions.

App-Separation Applications must be physically and/or logically separated from each other, such that no information is available between applications except as may be specifically intended. These applications should not share cryptographic keys

Rollback In the instance of interruption of an operation through power failure or premature withdrawal of the card, the TOE shall return all operational values to their status at the beginning of that operation.

De-allocation When an operation utilizes registers or temporary storage, that register or storage shall have all security critical information removed before the operation completes.

Application Application dependent information flow control provisions shall be fully described.

Other Additional information flow control SFP elements shall be specified as appropriate.

6.3 Management of Functions in TSF

Component FMT_MOF allows certain authorized roles to manage the behavior of functions in the TSF that use rules or have specified conditions that may be manageable. Table 6.1 lists the IT security functional components and indicates whether the component has a possible management function identified in the CC and if some or all of that function may be applied to the VSCPP compliant TOE. Following the table, each requirement is listed. Comments regarding exclusion are indicated in **bold** type. Management functions to be considered are indicated in ***bold italic*** type. In general, exclusions are a result of the limited memory and operational capability of the TOE coupled with the unique operational limitations discussed above.

Table 6.1 – Security Functional Components Management Options

Component	Component Name	Management Functions?	Actions Considered?
FCS_CKM.3	Cryptographic key access	yes	no
FCS_COP.1	Cryptographic operation	no	no
FDP_ACC.1	Subset access control	no	no
FDP_ACF.1	Security attribute based access control	yes	yes
FDP_ETC.1	Export of user data w/o security attributes	no	no
FDP_IFC.1	Subset information flow control	no	no
FDP_IFF.1	Simple security attributes	yes	yes
FDP_ITC.1	Import of user data w/o security attributes	yes	yes
FDP_ITT.1	Basic internal transfer protection	yes	no
FDP_RIP.2	Full residual information protection	yes	no
FDP_ROL.2	Advanced rollback	yes	no
FDP_SDI.2	Stored data integrity monitoring and action	yes	no
FDP_UIT.1	Data exchange integrity	no	no
FIA_AFL.1	Authentication failure handling	yes	yes
FIA_ATD.1	User attribute definition	yes	no
FIA_SOS.1	Verification of secrets	yes	no

Component	Component Name	Management Functions?	Actions Considered?
FIA_SOS.2	TSF Generation of secrets	yes	no
FIA_UAU.1	Timing of authentication	yes	yes
FIA_UAU.4	Single-use authentication mechanisms	no	no
FIA_UAU.5	Multiple authentication mechanisms	yes	yes
FIA_UAU.7	Protected authentication feedback	no	no
FIA_UID.1	Timing of identification	yes	no
FMT_MOF.1	Management of security functions behavior	yes	no
FMT_MSA.1	Management of security attributes	yes	no
FMT_MSA.2	Secure security attributes	no	no
FMT_MSA.3	Static attribute initialization	yes	no
FMT_MTD.1	Management of TSF data	yes	no
FMT_MTD.2	Management of limits on TSF data	yes	no
FMT_MTD.3	Secure TSF data	no	no
FMT_REV.1	Revocation	yes	yes
FPT_FLS.1	Failure with preservation of secure state	no	no
FPT_ITI.1	Inter-TSF detection of modification	no	no
FPT_ITT.1	Basic internal TSF data transfer protection	yes	no
FPT_PHP.3	Resistance to physical attack	yes	no
FPT_RCV.3	Automated recovery without undue loss	yes	no
FPT_RCV.4	Function recovery	no	no
FPT_RPL.1	Replay detection	yes	yes
FPT_RVM.1	Non-bypassability of the TSP	no	no
FPT_SEP.1	TSF domain separation	no	no
FPT_TST.1	TSF testing	yes	yes

FCS_CKM.3 - Cryptographic key access

The following actions could be considered for the management functions in FMT:

a) the management of changes to cryptographic key attributes. Examples of key attributes include user, key type (e.g. public, private, secret), validity period, and use (e.g. digital signature, key encryption, key agreement, data encryption).

Not applicable for VSCPP compliant TOEs.

FCS_COP.1 - Cryptographic operation

There are no management activities foreseen for these components.

FDP_ACC.1 - Subset access control

There are no management activities foreseen for this component.

FDP_ACF.1 - Security attribute based access control

The following actions could be considered for the management functions in FMT Management:

a) Managing the ***attributes used to make explicit access or denial based decisions***.

FDP_ETC.1 - Export of user data without security attributes

There are no management activities foreseen for this component.

FDP_IFC.1 - Subset information flow control

There are no management activities foreseen for this component.

FDP_IFF.1 - Simple security attributes

The following actions could be considered for the management functions in FMT Management:

a) Managing the ***attributes used to make explicit access based decisions***.

FDP_ITC.1 - Import of user data without security attributes

The following actions could be considered for the management functions in FMT Management:

a) The ***modification of the additional control rules used for import***.

FDP_ITT.1 - Basic internal transfer protection

The following actions could be considered for the management functions in FMT Management:

a) If the TSF provides multiple methods to protect user data during transmission between physically separated parts of the TOE, the TSF could provide a pre-defined role with the ability to select the method that will be used.

Not applicable for VSCPP compliant TOEs.

FDP_RIP.2 - Full residual information protection

The following actions could be considered for the management functions in FMT Management:

a) The choice of when to perform residual information protection (i.e. upon allocation or de-allocation) could be made configurable within the TOE.

Not applicable for VSCPP compliant TOEs.

FDP_ROL.2 - Advanced rollback

The following actions could be considered for the management functions in FMT Management:

a) The boundary limit to which rollback may be performed could be a configurable item within the TOE.

b) Permission to perform a rollback operation could be restricted to a well defined role.

Not applicable for VSCPP compliant TOEs.

FDP_SDI.2 - Stored data integrity monitoring and action

The following actions could be considered for the management functions in FMT Management:

a) The actions to be taken upon the detection of an integrity error could be configurable.

Not applicable for VSCPP compliant TOEs.

FDP_UIT.1 - Data exchange integrity

There are no management activities foreseen for this component.

FIA_AFL.1 - Authentication failure handling

The following actions could be considered for the management functions in FMT:

a) management of the ***threshold for unsuccessful authentication attempts***;

b) management of ***actions to be taken in the event of an authentication failure***.

FIA_ATD.1 - User attribute definition

The following actions could be considered for the management functions in FMT:

a) if so indicated in the assignment, the authorized administrator might be able to define additional security attributes for users.

Not applicable for VSCPP compliant TOEs.

FIA_SOS.1 - Verification of secrets

The following actions could be considered for the management functions in FMT:

a) the management of the metric used to verify the secrets.

Not applicable for VSCPP compliant TOEs.

FIA_SOS.2 - TSF Generation of secrets

The following actions could be considered for the management functions in FMT:

a) the management of the metric used to generate the secrets.

Not applicable for VSCPP compliant TOEs.

FIA_UAU.1 - Timing of authentication

The following actions could be considered for the management functions in FMT:

a) management of the ***authentication data by an administrator***;

b) management of the authentication data by the associated user;

c) managing the list of actions that can be taken before the user is authenticated.

FIA_UAU.4 - Single-use authentication mechanisms

There are no management activities foreseen.

FIA_UAU.5 - Multiple authentication mechanisms

The following actions could be considered for the management functions in FMT:

a) the management of authentication mechanisms;

b) the ***management of the rules for authentication***.

FIA_UAU.7 - Protected authentication feedback

There are no management activities foreseen.

FIA_UID.1 Timing of identification

The following actions could be considered for the management functions in FMT:

a) the management of the user identities;

b) if an authorized administrator can change the actions allowed before identification, the managing of the action lists.

Not applicable for VSCPP compliant TOEs.

FMT_MOF.1 - Management of security functions behavior

The following actions could be considered for the management functions in FMT Management:

a) managing the group of roles that can interact with the functions in the TSF;

Not applicable for VSCPP compliant TOEs.

FMT_MSA.1 - Management of security attributes

The following actions could be considered for the management functions in FMT Management:

a) managing the group of roles that can interact with the security attributes.

Not applicable for VSCPP compliant TOEs.

FMT_MSA.2 - Secure security attributes

There are no additional management activities foreseen for this component.

FMT_MSA.3 - Static attribute initialization

The following actions could be considered for the management functions in FMT Management:

a) managing the group of roles that can specify initial values;

b) managing the permissive or restrictive setting of default values for a given access control SFP.

Not applicable for VSCPP compliant TOEs.

FMT_MTD.1 - Management of TSF data

The following actions could be considered for the management functions in FMT Management:

a) managing the group of roles that can interact with the TSF data.

Not applicable for VSCPP compliant TOEs.

FMT_MTD.2 - Management of limits on TSF data

The following actions could be considered for the management functions in FMT Management:

a) managing the group of roles that can interact with the limits on the TSF data.

Not applicable for VSCPP compliant TOEs.

FMT_MTD.3 - Secure TSF data

There are no additional management activities foreseen for this component.

FMT_REV.1 - Revocation

The following actions could be considered for the management functions in FMT Management:

a) managing the group of roles that can invoke revocation of security attributes;

b) managing the lists of users, subjects, objects and other resources for which revocation is possible;

c) ***managing the revocation rules***.

FPT_FLS.1 - Failure with preservation of secure state

There are no management activities foreseen.

FPT_ITI.1 - Inter-TSF detection of modification

There are no management activities foreseen.

FPT_ITT.1 - Basic internal TSF data transfer protection

The following actions could be considered for the management functions in FMT:

a) management of the types of modification against which the TSF should protect;

b) management of the mechanism used to provide the protection of the data in transit between different parts of the TSF.

Not applicable for VSCPP compliant TOEs.

FPT_PHP.3 - Resistance to physical attack

The following actions could be considered for the management functions in FMT:

a) management of the automatic responses to physical tampering.

Not applicable for VSCPP compliant TOEs.

FPT_RCV.3 - Automated recovery without undue loss

The following actions could be considered for the management functions in FMT:

a) management of who can access the restore capability within the maintenance mode;

b) management of the list of failures/service discontinuities that will be handled through the automatic procedures.

Not applicable for VSCPP compliant TOEs.

FPT_RCV.4 - Function recovery

There are no management activities foreseen.

FPT_RPL.1 - Replay detection

The following actions could be considered for the management functions in FMT:

a) management of the list of identified entities for which replay shall be detected;

b) management of the ***list of actions that need to be taken in case of replay***.

FPT_RVM.1 - Non-bypassability of the TSP

There are no management activities foreseen.

FPT_SEP.1 - TSF domain separation

There are no management activities foreseen.

FPT_TST.1 - TSF testing

The following actions could be considered for the management functions in FMT:

a) management of the ***conditions under which TSF self testing occurs***, such as during initial start-up, regular interval, or under specified conditions;

b) management of the time interval if appropriate.

7 Rationale

7.1 Introduction and TOE Description Rationale

The Target of Evaluation, a smart card, has been defined. This TOE has a unique set of threats relating to its character as a small, self contained microprocessor that is powered only when connected to a reader, is manufactured in large quantities, and is issued to untrusted users for their long-term retention. The description of the TOE supports the statement of threats, policies, and assumptions discussed above. It also provides information sufficient to support application notes and the further development of the objectives and requirements.

7.2 Security Objectives Rationale

This section demonstrates that the stated security objectives counter all identified threats, policies and assumptions.

Table 7.1 Environmental Considerations Related to Objectives

Environmental Consideration	Is Addressed By Objective(s)
A.Attack	O.Phys_Prot, O.Log_Prot, O.I_Leak, O.Operate
A.User	O.DAC, O.FAC
A.Admin	O.Admin
T.P_Probe	O.Phys_Prot, O.D_Read, O.Operate
T.P_Modify	O.Phys_Prot, O.Operate
T.E_Manip	O.Phys_Prot
T.Us_Error	O.Log_Prot
T.UA_Op	O.Log_Prot
T.UA_Load	O.Log_Prot, O.I_Leak, O.Operate
T.Cmd_Str	O.Log_Prot
T.Forcd_Rst	O.Log_Prot, O.Init
T.Trns_Integ	O.Log_Prot, O.Sec_Com
T.Flt_Ins	O.Flt_Ins

Environmental Consideration	Is Addressed By Objective(s)
T.Re-Use	O.Re-Use
T.Load_Mal	O.Log_Prot, OE.Init_Acs
T.Priv	O.Log_Prot, OE.Perss
T.First_Use	O.Set_Up, O.Init
T.Impers	O.Set_Up
T.Access	O.DAC
T.Search	O.Log_Prot, O.Search
T.Cmd_Ftn	O.Log_Prot
T.Load_Flt	O.Log_Prot
T.SWBld_Fail	O.Flaw
T.HWBld_Fail	O.Flaw
T.Alt_Ftn	O.Log_Prot, O. Mult_App, O.Life_Cycle, O.Operate
T.Gen_Atk	O.Log_Prot
T.Crypt_Atk	O.Set_Up, O.Crypt
T.IO_Man	O.Log_Prot
T.I_Leak	O.Phys_Prot, O.I_Leak, O.D_Read
T.Link	O.Unlink
T.Lnk_Att	O.Log_Prot, O.Mem_Chk
T.Env_Strs	O.Phys_Prot
T.Dis_Des	OE.Con_Des, OE.Dlv_Proc, OE.Dlv_Aud, OE.Dlv_Trn, OE.Sampl_Acs
T.Dis_Soft	OE.Con_Des, OE.Dlv_Proc, OE.Dlv_Aud, OE.Dlv_Trn, OE.Sampl_Acs
T.Dis_Data	OE.Con_Des, OE.Dlv_Proc, OE.Dlv_Aud, OE.Dlv_Trn, OE.Sampl_Acs, OE.Init_Acs
T.Dis_Test	OE.Con_Des, OE.Dlv_Proc, OE.Dlv_Aud, OE.Dlv_Trn
T.Tft_Prod	OE.Con_Prod, OE.Dlv_Proc, OE.Dlv_Aud, OE.Dlv_Trn,

Environmental Consideration	Is Addressed By Objective(s)
	OE.Sampl_Acs
T.Tft_Mask	OE.Con_Prod, OE.Mask_Prot, OE.Dlv_Proc
T.Tft_Tools	OE.Con_Prod
T.Mod_Des	OE.Con_Des, OE.Dlv_Proc, OE.Dlv_Aud, OE.Dlv_Trn
T.Mod_Soft	OE.Con_Des, OE.Dlv_Proc, OE.Dlv_Aud, OE.Dlv_Trn
T.Mod_Data	OE.Con_Des, OE.Dlv_Proc, OE.Dlv_Aud, OE.Dlv_Trn, OE.Init_Acs
T.Mod_Test	OE.Con_Des, OE.Sampl_Acs
T.Key_Comp	OE.Key_Con
T.Clon	OE.Con_Des, OE.Con_Prod, OE.Mask_Prot, OE.Dlv_Proc, OE.Sampl_Acs, OE.Init_Acs
P.Data_Acc	O.DAC
P.File_Acc	O.FAC
P.Mult_App	O. Mult_App
P.Crypt_Std	O.Crypt
P.Ident	O.Ident
P.Sec_Com	O.Sec_Com
P.IT_Std	O.IT_Std
P.Extend	O.Extend
P.Con_Cont	OE.Con_Cont

Table 7.2 Security Objectives Related to Environmental Considerations

Security Objective	Is Necessitated By:
O.Phys_Prot	A.Attack , T.P_Probe, T.P_Modify, T.E_Manip, T.I_Leak, T.Env_Strs
O.Log_Prot	A.Attack, T.Us_Error, T.UA_Op, T.UA_Load, T.Cmd_Str, T.Forcd_Rst, T.Trns_Integ, T.Load_Mal, T.Priv, T.Search, T.Cmd_Ftn, T.Load_Flt, T.Alt_Ftn, T.IO_Man, T.Lnk_Att, T.Gen_Atk
O.DAC	A.User, T.Access, P.Data_Acc
O.FAC	A.User, P.File_Acc
O.I_Leak	A.Attack , T.UA_Load, T.I_Leak
O.Set_Up	T.First_Use, T.Impers, T.Crypt_Atk
O. Mult_App	T.Alt_Ftn, P.Mult_App
O.Life_Cycle	T.Alt_Ftn
O.Crypt	T.Crypt_Atk, P.Crypt_Std
O.Search	T.Search
O.Flt_Ins	T.Flt_Ins
O.Re-Use	T.Re-Use
O.Ident	P.Ident
O.Init	T.Forcd_Rst, T.First_Use
O.D_Read	T.P_Probe, T.I_Leak
O.Sec_Com	T.Trns_Integ, P.Sec_Com
O.Mem_Chk	T.Lnk_Att
O.Extend	P.Extend
O.Unlink	T.Link
O.Operate	A.Attack , T.P_Probe, T.P_Modify, T.UA_Load, T.Alt_Ftn
O.Flaw	T.SWBld_Fail, T.HWBld_Fail
O.Admin	A.Admin

Security Objective	Is Necessitated By:
O.IT_Std	P.IT_Std
OE.Con_Des	T.Dis_Des, T.Dis_Soft, T.Dis_Data, T.Dis_Test, T.Mod_Des, T.Mod_Soft, T.Mod_Data, T.Mod_Test, T.Clon
OE.Con_Prod	T.Tft_Prod, T.Tft_Mask, T.Tft_Tools, T.Clon
OE.Mask_Prot	T.Tft_Mask, T.Clon
OE.Dlv_Proc	T.Dis_Des, T.Dis_Soft, T.Dis_Data, T.Dis_Test, T.Tft_Prod, T.Tft_Mask, T.Mod_Des, T.Mod_Soft, T.Mod_Data, T.Clon
OE.Dlv_Aud	T.Dis_Des, T.Dis_Soft, T.Dis_Data, T.Dis_Test, T.Tft_Prod, T.Mod_Des, T.Mod_Soft, T.Mod_Data
OE.Dlv_Trn	T.Dis_Des, T.Dis_Soft, T.Dis_Data, T.Dis_Test, T.Tft_Prod, T.Mod_Des, T.Mod_Soft, T.Mod_Data
OE.Sampl_Acs	T.Dis_Des, T.Dis_Soft, T.Dis_Data, T.Tft_Prod, T.Mod_Test, T.Clon
OE.Init_Acs	T.Load_Mal, T.Dis_Data, T.Mod_Data, T.Clon
OE.Perss	T.Priv
OE.Key_Con	T.Key_Comp
OE.Con_Cont	P.Con_Cont

7.3 Security Requirements Rationale

This section provides the rationale for necessity and sufficiency of security requirements, demonstrating that each of the security objectives is addressed by at least one security requirement and that every security requirement is directed toward solving at least one objective.

Table 7.3 Security Objectives Related to Security Requirements

Security Objective	Is Addressed By:
O.Phys_Prot	FPT_PHP.3, ADV_IMP.1, ATE_FUN.1, AVA_VLA.3
O.Log_Prot	FDP_RIP.2, FDP_SDI.2, FIA_AFL.1, FIA_UAU.7, FMT_MSA.2, FMT_MTD.3, FPT_FLS.1, FPT_RCV.3, FPT_RCV.4, FPT_RVM.1, FPT_SEP.1, ATE_FUN.1, ADV_IMP.1, AVA_VLA.3
O.DAC	FDP_ACC.1, FDP_IFC.1, FIA_UAU.1, FIA_UAU.5, FIA_UID.1, FMT_MSA.1, FMT_MTD.1, FPT_TST.1, FMT_SMR.1
O.FAC	FDP_ACC.1, FIA_UAU.1, FIA_UID.1, FMT_SMR.1
O.I_Leak	FDP_IFF.1, AVA_VLA.3
O.Set_Up	FDP_ACC.1, FIA_UAU.5, FMT_MSA.2, FMT_MSA.3, ADV_IMP.1
O. Mult_App	FDP_ACC.1, FDP_IFC.1, ADV_IMP.1, ATE_DPT.1, ATE_FUN.1, AVA_VLA.3
O.Life_Cycle	ADV_IMP.1, AVA_VLA.3
O.Crypt	FCS_CKM.1, FCS_CKM.4, FCS_COP.1, FIA_SOS.1, FIA_SOS.2, FIA_UAU.4, AVA_SOF.1
O.Search	FIA_AFL.1
O.Flt_Ins	AVA_VLA.3
O.Re-Use	FPT_RPL.1
O.Ident	ATE_FUN.1
O.Init	FDP_RIP.2, FDP_ROL.2, FPT_RCV.3, FPT_RCV.4,

Security Objective	Is Addressed By:
	ADV_IMP.1, ATE_FUN.1, AVA_VLA.3
O.D_Read	FDP_ITT.1, FPT_ITT.1
O.Sec_Com	FDP_ETC.1, FDP_ITC.1, FDP_UIT.1, FIA_SOS.1, FIA_SOS.2, FIA_UAU.4, FPT_ITI.1, ATE_FUN.1
O.Mem_Chk	FDP_SDI.2
O.Extend	FMT_MOF.1
O.Unlink	AVA_VLA.3
O.Operate	FPT_FLS.1, FPT_RVM.1, FPT_SEP.1, ATE_COV.2, ATE_DPT.1, ATE_FUN.1, ATE_IND.2
O.Flaw	ACM_AUT.1, ACM_CAP.4, ACM_SCP.2, ADV_HLD.2, ADV_INT.1, ADV_LLD.1, ADV_RCR.1, ADV_SPM.1, ALC_LCD.1, ALC_TAT.1, ATE_COV.2, ATE_DPT.1, ATE_FUN.1, ATE_IND.2
O.Admin	FDP_ACF.1, FIA_ATD.1, FMT_MOF.1, FMT_MSA.1, FMT_MSA.3, FMT_MTD.2, FMT_REV.1
O.IT_Std	FCS_CKM.3, FCS_COP.1, ADV_FSP.2, ADV_HLD.2
OE.Con_Des	ALC_DVS.1
OE.Con_Prod	ADO_DEL.2, ALC_DVS.1
OE.Mask_Prot	ALC_DVS.1
OE.Dlv_Proc	ADO_DEL.2, ADO_IGS.1, ALC_DVS.1
OE.Dlv_Aud	ADO_DEL.2, ALC_DVS.1
OE.Dlv_Trn	ADO_DEL.2, ALC_DVS.1
OE.Sampl_Acs	ALC_DVS.1
OE.Init_Acs	ALC_DVS.1
OE.Perss	AGD_ADM.1, AVA_MSU.2
OE.Key_Con	AGD_USR.1
OE.Con_Cont	ACM_CAP.4

Table 7.4 Security Requirements Related to Security Objectives

Security Requirement	Is Necessitated By:
FCS_CKM.1	O.Crypt
FCS_CKM.3	O.IT_Std
FCS_CKM.4	O.Crypt
FCS_COP.1	O.Crypt, O.IT_Std
FDP_ACC.1	O.DAC, O.FAC, O.Set_Up, O. Mult_App
FDP_ACF.1	O.Admin
FDP_ETC.1	O.Sec_Com
FDP_IFC.1	O.DAC, O. Mult_App
FDP_IFF.1	O.I_Leak
FDP_ITC.1	O.Sec_Com
FDP_ITT.1	O.D_Read
FDP_RIP.2	O.Log_Prot, O.Init
FDP_ROL.2	O.Init
FDP_SDI.2	O.Log_Prot, O.Mem_Chk
FDP_UIT.1	O.Sec_Com
FIA_AFL.1	O.Log_Prot, O.Search
FIA_ATD.1	O.Admin
FIA_SOS.1	O.Crypt, O.Sec_Com
FIA_SOS.2	O.Crypt, O.Sec_Com
FIA_UAU.1	O.DAC, O.FAC
FIA_UAU.4	O.Crypt, O.Sec_Com
FIA_UAU.5	O.DAC, O.Set_Up
FIA_UAU.7	O.Log_Prot
FIA_UID.1	O.DAC, O.FAC
FMT_MOF.1	O.Extend, O.Admin

Security Requirement	Is Necessitated By:
FMT_MSA.1	O.DAC, O.Admin
FMT_MSA.2	O.Log_Prot, O.Set_Up
FMT_MSA.3	O.Set_Up, O.Admin
FMT_MTD.1	O.DAC
FMT_MTD.2	O.Admin
FMT_MTD.3	O.Log_Prot
FMT_REV.1	O.Admin
FMT_SMR.1	O.DAC, O.FAC
FPT_FLS.1	O.Log_Prot, O.Operate
FPT_ITI.1	O.Sec_Com
FPT_ITT.1	O.D_Read
FPT_PHP.3	O.Phys_Prot
FPT_RCV.3	O.Log_Prot, O.Init
FPT_RCV.4	O.Log_Prot, O.Init
FPT_RPL.1	O.Re-Use
FPT_RVM.1	O.Log_Prot, O.Operate
FPT_SEP.1	O.Log_Prot, O.Operate
FPT_TST.1	O.DAC
ACM_AUT.1	O.Flaw
ACM_CAP.4	O.Flaw, OE.Con_Cont
ACM_SCP.2	O.Flaw
ADO_DEL.2	OE.Con_Prod, OE.Dlv_Proc, OE.Dlv_Aud, OE.Dlv_Trn
ADO_IGS.1	OE.Dlv_Proc
ADV_FSP.2	O.IT_Std
ADV_HLD.2	O.IT_Std, O.Flaw
ADV_IMP.1	O.Phys_Prot, O.Log_Prot, O.Set_Up, O. Mult_App, O.Life_Cycle, O.Init

Security Requirement	Is Necessitated By:
ADV_INT.1	O.Flaw
ADV_LLD.1	O.Flaw
ADV_SPM.1	O.Flaw
ADV_RCR.1	O.Flaw
AGD_ADM.1	OE.Perss
AGD_USR.1	OE.Key_Con
ALC_DVS.1	OE.Con_Des, OE.Con_Prod, OE.Mask_Prot, OE.Dlv_Proc, OE.Dlv_Aud, OE.Dlv_Trn, OE.Sampl_Acs, OE.Init_Acs
ALC_LCD.1	O.Flaw
ALC_TAT.1	O.Flaw
ATE_COV.2	O.Operate, O.Flaw
ATE_DPT.1	O. Mult_App, O.Operate, O.Flaw
ATE_FUN.1	O.Phys_Prot, O.Log_Prot, O. Mult_App, O.Ident, O.Init, O.Sec_Com, O.Operate, O.Flaw
ATE_IND.2	O.Operate, O.Flaw
AVA_MSU.2	OE.Perss
AVA_SOF.1	O.Crypt
AVA_VLA.3	O.Phys_Prot, O.Log_Prot, O.I_Leak, O. Mult_App, O.Life_Cycle, O.Flt_Ins, O.Init, O.Unlink

7.4 Rationale that Dependencies are Satisfied

The selected security requirements include related dependencies that must be met or their exclusion justified. The following table provides this information.

Table 7.5 Summary of Dependencies

Component	Depends On	Which is
FCS_CKM.3	FCS_CKM.1	Included
FCS_CKM.3	FCS_CKM.4	Included
FCS_CKM.3	FDP_ITC.1	Included
FCS_CKM.3	FMT_MSA.2	Included
FCS_COP.1	FCS_CKM.1	Included
FCS_COP.1	FCS_CKM.4	Included
FCS_COP.1	FDP_ITC.1	Included
FCS_COP.1	FMT_MSA.2	Included
FDP_ACC.1	FDP_ACF.1	Included
FDP_ACF.1	FDP_ACC.1	Included
FDP_ACF.1	FMT_MSA.3	Included
FDP_ETC.1	FDP_ACC.1	Included
FDP_ETC.1	FDP_IFC.1	Included
FDP_IFC.1	FDP_IFF.1	Included
FDP_IFF.1	FDP_IFC.1	Included
FDP_IFF.1	FMT_MSA.3	Included
FDP_ITC.1	FDP_ACC.1	Included
FDP_ITC.1	FDP_IFC.1	Included
FDP_ITC.1	FMT_MSA.3	Included
FDP_ITT.1	FDP_ACC.1	Included
FDP_ITT.1	FDP_IFC.1	Included
FDP_ROL.2	FDP_ACC.1	Included

Component	Depends On	Which is
FDP_ROL.2	FDP_IFC.1	Included
FDP_UIT.1	FDP_ACC.1	Included
FDP_UIT.1	FDP_IFC.1	Included
FDP_UIT.1	FTP_ITC.1	Not Included - See below
FDP_UIT.1	FTP_TRP.1	Not Included - See below
FIA_AFL.1	FIA_UAU.1	Included
FIA_UAU.1	FIA_UID.1	Included
FIA_UAU.7	FIA_UAU.1	Included
FMT_MOF.1	FMT_SMR.1	Included
FMT_MSA.1	FDP_ACC.1	Included
FMT_MSA.1	FDP_IFC.1	Included
FMT_MSA.1	FMT_SMR.1	Included
FMT_MSA.2	FDP_ACC.1	Included
FMT_MSA.2	FDP_IFC.1	Included
FMT_MSA.2	FMT_MSA.1	Included
FMT_MSA.2	FMT_SMR.1	Included
FMT_MSA.2	ADV_SPM.1	Included
FMT_MSA.3	FMT_MSA.1	Included
FMT_MSA.3	FMT_SMR.1	Included
FMT_MTD.1	FMT_SMR.1	Included
FMT_MTD.2	FMT_MTD.1	Included
FMT_MTD.2	FMT_SMR.1	Included
FMT_MTD.3	FMT_MTD.1	Included
FMT_MTD.3	ADV_SPM.1	Included
FMT_REV.1	FMT_SMR.1	Included
FPT_FLS.1	ADV_SPM.1	Included
FPT_RCV.3	FPT_TST.1	Included
FPT_RCV.3	ADV_SPM.1	Included

Component	Depends On	Which is
FPT_RCV.3	AGD_ADM.1	Included
FPT_RCV.4	ADV_SPM.1	Included
FPT_TST.1	FPT_AMT.1	Not Included - See below
ACM_AUT.1	ACM_CAP.4	Included
ACM_CAP.4	ALC_DVS.1	Included
ACM_SCP.2	ACM_CAP.3	Included
ADO_DEL.2	ACM_CAP.3	Included
ADO_IGS.1	AGD_ADM.1	Included
ADV_FSP.2	ADV_RCR.1	Included
ADV_HLD.2	ADV_RCR.1	Included
ADV_IMP.1	ADV_LLD.1	Included
ADV_IMP.1	ADV_RCR.1	Included
ADV_IMP.1	ALC_TAT.1	Included
ADV_INT.1	ADV_IMP.1	Included
ADV_INT.1	ADV_LLD.1	Included
ADV_LLD.1	ADV_HLD.2	Included
ADV_LLD.1	ADV_RCR.1	Included
ALC_TAT.1	ADV_IMP.1	Included
ATE_COV.2	ATE_FUN.1	Included
ATE_DPT.1	ATE_FUN.1	Included
ATE_IND.2	AGD_ADM.1	Included
ATE_IND.2	AGD_USR.1	Included
ATE_IND.2	ATE_FUN.1	Included
AVA_MSU.2	ADO_IGS.1	Included
AVA_MSU.2	AGD_ADM.1	Included
AVA_MSU.2	AGD_USR.1	Included
AVA_VLA.3	ADV_HLD.2	Included
AVA_VLA.3	ADV_IMP.1	Included

Component	Depends On	Which is
AVA_VLA.3	ADV_LLD.1	Included
AVA_VLA.3	AGD_ADM.1	Included
AVA_VLA.3	AGD_USR.1	Included

Justification of Unsupported Dependencies

Dependency of FPT_TST.1 on FPT_AMT.1 –Abstract Machine Testing

The smart card TOE depends on the card reader device for support of all interactions. The inherent security of the TOE must, however, be strictly resident inside the TOE itself. Testing of the abstract machine is therefore not appropriate. Note, however, that the TOE may impose specific requirements on data control through challenge-response operations.

Dependency of FDP_UIT.1 on FTP_ITC.1 –Inter-TSF trusted channel

Smart cards are inserted to a card reader device for operation. The reader may be as simple as a small hand-held battery powered balance checker (for financial applications) or as complex as a large, highly secure cabinet with multiple levels of control. Because of this potential range of communication capabilities, it is inappropriate to require trusted channels. The TOE must maintain its own control.

Dependency of FDP_UIT.1 on FTP_TRP.1 –Trusted path

Smart cards are inserted into a card reader device for operation. The reader may be as simple as a small hand-held battery powered balance checker (for financial applications) or as complex as a large, highly secure cabinet with multiple levels of control. Because of this potential range of communication capabilities, it is inappropriate to require trusted paths. The TOE must maintain its own control.

7.5 Rationale for Strength of Function Medium

Component AVA_SOF provides for a qualification of claims of strength of function of security mechanisms. The claims and evaluation methodologies are defined in the supporting CC evaluation methodology. Annex B of the draft Common Evaluation Methodology (CEM) Version 0.6, 99/008, January 1999, details these claims, indicating that an SOF –high rating would provide adequate protection against an attacker with a high attack potential. Further definition specifies that this would require an Expert using Specialized ("controlled, possibly even restricted") equipment Months or Years to launch a successful attack. As there is no "specialized" equipment (as defined in the CEM) relevant to these attacks, no smart card could be evaluated as possessing an SOF- high rating.

In addition to the limitations on testability and the ability of an evaluation facility to properly review this claim, it is problematical that such resistance could be proven. It only takes months for an efficient integrated circuit lab to reverse engineer a microprocessor (a far more complex integrated circuit than that used in smart cards). This is routinely accomplished by IC manufacturers and their competitors. It can be expected that the effort necessary to successfully defeat the TOE referenced here would be considerably less.

A strength of function –medium rating is therefore justified on practicality, cost effectiveness, and efficiency.

7.6 Rationale for Assurance Level EAL4 Augmented

The assurance level for this Protection Profile is EAL4 augmented.

EAL4 allows a developer to attain a reasonably high assurance level without the need for highly specialized processes and practices. It is considered to be the highest level that could be applied to an existing product line without undue expense and complexity. As such, EAL4 is appropriate for commercial products which can be applied to moderate to high security functions. Smart cards are just such a product.

Augmentation results from the selection of:

AVA_VLA.3 Vulnerability Assessment; Vulnerability Analysis; Moderately resistant

The rationale for this claim is based on the CEM definitions of basic/medium/high attack potentials. These definitions apply most directly to large information processing systems which exist in small numbers and are offered some form of external protection. Smart cards, as discussed above, are issued in large quantities, are exposed for prolonged periods of time and are subject to short duration secondary attacks based on longer term development of sophisticated capabilities. As a result, the attack potentials, as stated, are not appropriate. They need to be redefined in this context for smart cards. With that understanding, a moderate attack potential would address the most reasonably expected competent attacks. Addressing all attacks at all levels (e.g., VLA.4) introduces cost and complexity higher than justified for all but the most secure applications. It is also questionable if this level can be achieved.

And

ADV_INT.1 Development; TSF internals; Modularity

With the rationale that the smart card TOE is composed of a collection of hardware and software functions that range from basic operating functions to advanced applications. These may be developed by one or a number of suppliers. As a result, it is important that the operations contained in the final product have the minimum possibility of destructive interaction. Imposing a requirement on modularity and elimination of unnecessary interactions supports this requirement.

Annex A –Glossary

This section contains only those terms which are used in a specialized way in the CC. The majority of terms in the CC are used either according to their accepted dictionary definitions or according to commonly accepted definitions that may be found in ISO security glossaries or other well-known collections of security terms.

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

Assignment The specification of an identified parameter in a component.

Assurance Ground for confidence that an entity meets its security objectives.

Attack potential The perceived potential for success of an attack, should an attack be launched, expressed in terms of an attacker's expertise, resources and motivation.

Augmentation The addition of one or more assurance component(s) from Part 3 to an EAL or assurance package.

Authentication data Information used to verify the claimed identity of a user.

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

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

Dependency A relationship between requirements such that the requirement that is depended upon must normally be satisfied for the other requirements to be able to meet their objectives.

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

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.

Human user Any person who interacts with the TOE.

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

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

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

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

Organizational security policies One or more security rules, procedures, practices, or guidelines imposed by an organization upon its operations.

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

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 authorized 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 organization 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.

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

SOF-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 organized breach of TOE security by attackers possessing a high attack potential.

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

Target of Evaluation (TOE) An IT product or system 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.