

# **TCPA TPMPP**

**Version 0.45**

**Thu Sep 14 16:08:30 PDT 2000**

**Prepared By: David Grawrock**

**Prepared For: T CPA membership**

**Copyright © 2000 Compaq Computer Corporation, Hewlett-Packard Company, IBM Corporation,  
Intel Corporation, Microsoft Corporation**

**All rights reserved.**

**DISCLAIMERS:**

**THIS SPECIFICATION IS PROVIDED “AS IS” WITH NO WARRANTIES WHATSOEVER INCLUDING ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION, OR SAMPLE.**

**NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED OR INTENDED HEREBY.**

**COMPAQ, HP, IBM, INTEL, AND MICROSOFT, DISCLAIM ALL LIABILITY, INCLUDING LIABILITY FOR INFRINGEMENT OF PROPRIETARY RIGHTS, RELATING TO THE USE OF THE INFORMATION IN THIS SPECIFICATION AND TO THE IMPLEMENTATION OF INFORMATION IN THIS SPECIFICATION. COMPAQ, HP, IBM, INTEL, AND MICROSOFT, DO NOT WARRANT OR REPRESENT THAT SUCH IMPLEMENTATION(S) WILL NOT INFRINGE SUCH RIGHTS.**

**WITHOUT LIMITATION, COMPAQ, HP, IBM, INTEL, AND MICROSOFT DISCLAIM ALL LIABILITY FOR COST OF PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES, LOST PROFITS, LOSS OF USE, LOSS OF DATA OR ANY INCIDENTAL, CONSEQUENTIAL, DIRECT, INDIRECT, OR SPECIAL DAMAGES, WHETHER UNDER CONTRACT, TORT, WARRANTY OR OTHERWISE, ARISING IN ANY WAY OUT OF USE OR RELIANCE UPON THIS SPECIFICATION OR ANY INFORMATION HEREIN.**

**All product names are trademarks, registered trademarks, or service marks of their respective owners.**

# Foreword

Before the Table of Contents there should be a foreword stating:

- a) the reason for the PP,
- b) where to send comments for the PP,
- c) a revision history of the PP,
- d) what authority the PP has, if any (e.g., A DoD standard, an NSA standard, an ANSI standard) and what communities (if any) are referencing the PP.

# Table Of Contents

— Item

## **1 - Introduction**

### **1.1 - Identification**

### **1.2 - Protection Profile Overview**

### **1.3 - Organisation (Optional)**

### **1.4 - Related Protection Profiles**

## **2 - TOE Description**

## **3 - TOE Security Environment**

### **3.1 - Secure Usage Assumptions**

### **3.2 - Threats to Security**

### **3.3 - Organisational Security Policies**

## **4 - Security Objectives**

### **4.1 - Security Objectives for the TOE**

### **4.2 - Security Objectives for the Environment**

## **5 - IT Security Requirements**

### **5.1 - TOE Security Functional Requirements**

## **5.2 - TOE Security Assurance Requirements**

## **5.3 - Security Requirements for the IT Environment (Optional)**

## **5.4 - Security Requirements for the Non-IT Environment (Optional)**

## **6 - Rationale**

### **6.1 - Introduction and TOE Description Rationale (Optional)**

### **6.2 - Security Objectives Rationale**

#### **6.2.1 - Policies**

#### **6.2.2 - Threats**

### **6.3 - Security Requirements Rationale**

#### **6.3.1 - Functional Security Requirements Rationale**

#### **6.3.2 - Assurance Security Requirements Rationale**

### **6.4 - Dependency Rationale**

### **6.5 - Security Functional Requirements Grounding in Objectives**

### **6.6 - Rationale for Extensions**

# **List of Tables**

— Item

**Table - 5-1 Assurance Requirements: EAL(3)**

**Table - 6-1 Mapping the TOE Security Environment to Security Objectives**

**Table - 6-2 Tracing of Security Objectives to the TOE Security Environment**

**Table - 6-3 Functional Component to Security Objective Mapping**

**Table - 6-4 Functional and Assurance Requirements Dependencies**

Table - 6-5 Requirements to Objectives Mapping

# Conventions and Terminology

## Conventions

Provide a description of any unique conventions to this document.

## Terminology

Security Attributes – The security attributes are:

- Migration – The migration attribute determines if the entity can migrate from one TPM to another.
- Creation – The creation attribute indicates the creation status of an entity. Internal indicates that the entity creation occurred in the TPM external indicates that the entity came from outside the TPM.
- Type – The type attribute indicates if the entity is a signature key, encryption key or storage entity.

# Document Organisation

Section 1 provides the introductory material for the protection profile

Section 2 provides general purpose and TOE description

Section 3 provides a discussion of the expected environment for the TOE. This section also defines the set of threats that are to be addressed by either the technical countermeasures implemented in the TOE hardware or software or through the environmental controls.

Section 4 defines the security objectives for both the TOE and the TOE environment.

Section 5 contains the functional and assurance requirements derived from the Common Criteria, Part 2 and 3, respectively, that must be satisfied by the TOE.

Section 6 provides a rationale to explicitly demonstrate that the information technology security objectives satisfy the policies and threats. Arguments are provided for the coverage of each policy and threat. The section then explains how the set of requirements are complete relative to the objectives, and that each security objective is addressed by one or more component requirements. Arguments are provided for the coverage of each objective. Next Section 6 provides a set of arguments that address dependency analysis, strength of function issues, and the internal consistency and mutual supportiveness of the protection profile requirements

An acronym list is provided to define frequently used acronyms.

A reference section is provided to identify background material.

## 1 - Introduction

---

Provide document management and overview information necessary to operate a protection profile registry. The Introduction should provide background information that enables the reader to gain a high-level understanding of the protection profile.

## 1.1 - Identification

Version Number: Draft Version 0.45

Registration:

A glossary of terms used in the protection profile (PP) is given in Annex B. This protection profile is hereafter referred to as the Trusted Computing Platform Alliance Trusted Platform Module Protection Profile (TCPA-TPMPP)

This PP has been built with Common Criteria (CC) Version 2.1 (ISO/IEC 15408) and Common Methodology for Information Technology Security Evaluation (CEM) 99/008 Version 0.6 January 1999.

The structure for this PP uses the Common Criteria Toolbox (Version 5.0i, 16 Feb 2000). This toolbox was developed by SPARTA, Inc., for the US National Security Agency. It is available through <http://cctoolbox.sparta.com>.

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

## 1.2 - Protection Profile Overview

This PP describes the IT security requirements for a security module known as the Trusted Platform Module (TPM). The TPM requires support from BIOS and software. The name in use for the supporting BIOS and software is Trusted Platform Subsystem (TPS). The security requirements in this PP apply to the TPM as shipped from the final assembly point of the TPM. The requirements in the TPMPP cover any hardware devices and firmware that create a TPM. There is no coverage for any entities in the TPS.

A separate PP known as the TCPA Trusted Platform Subsystem Protection Profile (TCPA-TPSPP) provides coverage for the TPS.

This PP does not target specific applications that use the TPM. Application coverage is at the operating system level would require a PP that provides coverage for an operating system.

The TPM is a collection of hardware and software. The mixture of hardware and software is a design feature for the manufacturer. The TCPA specification identifies "protected capabilities" and "protected data". The TPM is the module that provides the functionality and storage for the protected capabilities and protected data. All other features and data are part of the TPS. The TPMPP defines what is appropriate protection and how to evaluate the protections in the manufacturers design. The goal is to allow purchasers of various implementations of a TPM to compare the products using the same criteria.

The TPM will be the TOE. The TPM in all implementations will require significant help from the environment to properly provide a complete security solution. In particular, the TPM requires significant help in authentication and resource utilization.

The TPM provides security primitives in a secure environment. The primitives will include digital signatures, random number generation, protected storage and binding information to the TPM.

While the TPMPP makes no requirements on having hardware protections in the creation of a TPM subsystem, if hardware protections are not present then sufficient protections in the

environment must be present. For example, if the TPM were a software module of an A1 rated system (using the old Rainbow series) then the protection provided by the environment might meet the requirements.

## TPM example

This definition is an example of one method of creating a TPM. Manufacturers may pick different ways of creating a TPM that are still compliant with the TPMPP.

For hardware TPM device, the TPM may be a computer chip embedded into a carrier. The chip is a semiconductor (silicon) integrated circuit (IC) fabricated in a complex microelectronic process, which involves repeatedly masking and doping the surface of a silicon substrate to form transistors, followed by patterning metal connections, and applying a protective overcoat. This process eventually yields a design comprising typically several hundred thousand transistors. The design consists of a central processing unit, an optional coprocessor, input and output lines, and volatile and non-volatile memory.

The chip will also be designed to be secure. In order to be secure, it should make appropriate use of both specific security enforcing design features, e.g. environmental sensors, and also technological properties of the materials and processes used.

A part of the manufacturing process is the inclusion of operating system (OS) developer-specific code, written in the microprocessor's native or machine code. This is usually contained in one of the numerous masks used during manufacture, referred to in this document as the ROM mask.

## Requirement Summary

### Functionality

The T CPA-TPMPP system targets these users needs –

- Providing a source of random numbers
- Providing a secure mechanism to perform the protected capability operations
- Providing a secure area to store the shielded data
- Providing mechanisms to store keys (both symmetric and asymmetric) that are useable only by the TPM
- Providing mechanisms to store integrity metrics
- Providing mechanisms to report on the integrity metrics
- Providing mechanisms to bind information to the TPM
- Providing mechanisms to manage the subsystem
- Providing resistance to resource depletion by providing resource allocation features
- Providing mechanisms to detect some insecurity
- Providing mechanisms for trusted recovery in the event of some system failures or detected insecurities
- Supporting these capabilities in a distributed system connected via an untrusted network

The T CPA-TPMPP is not expected to require that the TOE –

- Adequately protect against malicious abuse of authorized privileges.

- Adequately protect against sophisticated attacks (to include denial-of-service).
- Adequately protect against sophisticated hardware attacks (chip peeling etc.).
- Provide sufficient protection against installation, operation or administration errors.

## **Assurance**

The TPMPP assurances are to provide a level of confidence resulting from existing best known methods of hardware and software development and no extensive third-party evaluation.

## **Assurance Level**

The assurance level for this protection profile is EAL3.

Strength of function is medium.

## **Related Standards and Documents**

- 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

## **Related Protection Profiles and Documents**

## **PP Organization**

The main sections of the PP are the TOE (target of evaluation) Description, TOE Security Environment, Security Objectives, IT Security Requirements, and Rationale.

The TOE Description provides general information about the TOE, serves as an aid to understanding its security requirements, and provides context for the PP's evaluation.

The TOE Security Environment describes security aspects of the environment in which the TOE is to be used and the manner in which it is to be employed. The TOE security environment includes:

- a) Assumptions regarding the TOE's intended usage and environment of use
- b) Threats relevant to secure TOE operation
- c) Organizational security policies with which the TOE must comply

The security objectives reflect the stated intent of the PP. They pertain to how the TOE will counter identified threats and how it will cover identified organizational security policies and assumptions. Each security objective is categorized as being for the TOE or for the environment.

The Security Requirements section provides detailed requirements, in separate subsections, for the TOE and its environment.

The IT security requirements are subdivided as follows:

- a) TOE Security Functional Requirements
- b) TOE Security Assurance Requirements



The Application notes contain additional supporting information on issues unique to smart cards, consideration of management functions, and suggestions for the application of this PP through the use of packages applying to the basic chip, operating software, and integrated platform.

The Rationale presents evidence that the PP is a complete and cohesive set of requirements and that a conformant TOE would provide an effective set of IT security countermeasures within the security environment.

The Rationale is in two main parts. First, a Security Objectives Rationale demonstrates that the stated security objectives are traceable to all of the aspects identified in the TOE security environment and are suitable to cover them. Then, a Security Requirements Rationale demonstrates that the security requirements (TOE and environment) are traceable to the security objectives and are suitable to meet them.

## **2 - TOE Description**

### **Overview**

The target of evaluation (TOE) is the integrated circuit and operating software, including the BIOS, of Trusted Platform Module (TPM). The security requirements in this protection profile apply to the TPM and its software, from its manufacture, to its delivery to the motherboard manufacturer, to the platform manufacturer and finally to the end user.

### **Definition of TOE**

Trusted Platform Module in this PP applies to the mechanism that provides the protected capabilities and shielded locations. The TPM could be an integrated circuit (IC) with non-volatile memory and a microprocessor or it could be a software module in a high assurance environment.

The TOE for the TPM must provide the assurances that the protected capabilities work properly and that the shielded locations properly shield the data.

### **Technology**

There are a variety of technology issues relating to the TPM, which must be clearly delineated in a TCPA-TPMPP compliant product. These include implementation of security functions, attachment to motherboard and non-volatile memory.

### **Security functions**

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.

### **Motherboard attachment**

The TPM device must attach to the motherboard in some manner. The exact manner of attachment is left to the motherboard manufacturer. The manner of attachment must be reported

to any challenger when the TPM is in operation. This allows the challenger to make a trust decision based on how the TPM is attached to the motherboard.

## **Non-volatile memory**

The type and amount of non-volatile memory available in the IC of the TPM must be reportable in a verifiable manner.

## **Cryptography**

A variety of cryptographic keys are in use with the TPM, including endorsement keys, identity keys, transport keys and wrapping keys. Handling of these keys must be done in accordance with the key management procedures of this PP.

Cryptography may be implemented in hardware or software, with various algorithms and various key lengths. Some cryptographic operations must be performed on the IC with no secret data ever leaving the IC package. Other operations may perform operations with secret data off the IC package and in the software stack.

Any TOE claiming compliance with this protection profile must handle cryptographic functions in accordance with applicable international, industrial or organizational policies.

## **Required security functionality**

TCPA-TPMPP specifies the requirements for a system with the security functionality listed below.

- Generating a random number
- Digitally signing a supplied value
- Storing a key
- Binding information to the subsystem
- Collecting integrity metrics
- Responding to requests regarding the state of the integrity metrics
- Auditing in support of individual accountability and detection of and response to insecurity
- Resource allocation features providing a measure of resistance to resource depletion
- Mechanisms for detecting some insecurities
- System recovery features providing a measure of survivability in the face of system failures and insecurities
- Automated support to help in the verification of secure delivery, installation, operation, and administration

## **Environments**

The TPM environment is highly variable. In general, the TPM is assumed to be in an uncontrolled environment with no guarantee of the TPM's physical security. This allows TPM devices to be in mobile devices that are left in hotel rooms. The addition of physical security to the TPM device, on a server in a locked room, adds to the trust a challenger can make regarding the TPM

## Attacker capabilities

Attackers are assumed to have various levels of expertise, resources, and motivation. Relevant expertise may be in general semiconductor technology, software engineering, hacking techniques or the specific TOE. Resources may range from personal computers 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. It is assumed that given sufficient time and expertise, any TPM can be compromised.

## TOE identification

Through selection of the ACM Configuration Management Class of assurance functions this PP imposes the requirement that a unique reference be utilized to ensure that there is no ambiguity in terms of which instance of the TOE is being evaluated. Labeling the TOE with this reference then ensures that users of the TOE can be aware of which instance of the TOE they are using. The TOE described herein is, however, a combination of hardware and software, each portion of which may be composed of a further collection of components. This aggregate collection offers the potential for confusion in identifying a unique reference for the TOE.

To further complicate identification, commonly an IC can be produced with multiple features, only some of which are enabled. The design layout of the IC (the photomask) determines the functionality; however, as fabrication technology improves, the identical design may be used to produce an otherwise identical chip but with a reduced feature size. Likewise, software features may be selectively employed, depending on hardware functions. However, the presence or absence of specific features may directly contribute to the possible introduction of vulnerabilities. For example, the size of the IC features is directly related to the relative difficulty of probing. A potentially unknown, but present, software feature may allow backdoors or other routes for penetration.

It is therefore essential that the unique reference for the TOE compliant with this PP allow the identification of at least

- The microprocessor specification
- The memory size and allocation (ROM, EEPROM, RAM, etc.)
- The physical instantiation of the IC design regarding layout and feature size
- All hardware security features on the IC, whether they are initially enabled or not
- All enabled hardware security features
- The connection of the TPM to the motherboard
- The software specification
- All software security features present, whether they are initially enabled or not
- All enabled software security features

The TOE Description is a critical part of the protection profile. Provide a TOE description that enables the reader to:

- (1) gain an understanding of how the system operates,
- (2) know where the component fits into the system, and

the relationship of the TOE elements or shows the relationship of the TOE to its environment.

## 3 - TOE Security Environment

Summarise the security environment in which the TOE will be used and the manner the TOE will be employed.

### 3.1 - Secure Usage Assumptions

A.Application\_use: TOE application use

The TOE will be in use in various applications (i.e. data storage, binding to platform, system integrity and others) to provide security services and protect sensitive information.

A.Configuration: TPM configuration

The TOE will be properly installed and configured.

A.Conformance: Conformance guarantee

The use of the TOE does not guarantee the security of the overall system. The TPM is a subsystem and requires significant support from the environment to provide a total security solution. The platform manufacturer and end user must provide the environmental security appropriate to the data functions that must be protected.

A.Hostile\_User: Hostile users

Users cannot be trusted and are considered to be hostile.

A.Outsider\_Med: Proficient threat agents

The TOE is subject to deliberate attack by threat agents proficient in the security behavior of the system.

A.Physical: Physical attack

The TOE is assumed to be protected from physical tampering through features and technologies.

A.System: System description

The TOE is assumed to be a subsystem of a computing platform that provides OS, storage, input/output to TOE and TOE utilization functions.

A.TCPAIdentityCertification: Identity Certification

The TOE requires outside help to certify identities.

A.TCPARootMeasurement: TCPA Root of measurement trust  
The TPS will supply the root of measurement trust.

A.Trusted\_User: Trusted users  
Authorized users are trusted not to compromise security.

## **3.2 - Threats to Security**

T.Admin\_Err\_Commit: Administrative errors of commission  
An administrator commits errors that directly compromise organizational security objectives or change the technical security policy enforced by the system or application.

T.Admin\_Err\_Omit: Administrative errors of omission  
The system administrator fails to perform some function essential to security.

T.Admin\_UserPriv: Administrator violates user privacy policy  
An administrator learns the identity (or other privacy related information) of user(s) in violation of user privacy policy. Privacy-related information is sensitive information associated with the identity of a user.

T.Component\_Failure: A critical system component fails  
Failure of one or more system components results in the loss of system-critical functionality.

T.Dev\_Flawed\_Code: Software containing security-related flaws  
A system or applications developer delivers code that does not perform according to specifications or contains security flaws.

T.EndorseExpose: Exposure of endorsement key  
The endorsement key provides the root of reporting trust. If exposed it provides the attacker numerous mechanisms that allow for the forgery, cloning and masquerading as a valid TPM

T.Failure\_DS\_Comp: Failure of a distributed system component  
Failure of a component that is part of a distributed system will cause other parts of the distributed system to malfunction or provide unreliable results.

T.GlobalSecret: Global secret exposure  
If the TOE has a global secret known to all TOE's then exposure of one TOE exposes all TOE's.

T.Hack\_AC: Hacker undetected system access  
A hacker gains undetected access to a system due to missing, weak and/or incorrectly implemented access control causing potential violations of integrity, confidentiality, or availability.

**T.Hack\_AvL\_Resource:** Hacker attempts resource denial of service  
A hacker executes commands, sends data, or performs other operations that make system resources unavailable to system users. Resources that may be denied to users include bandwidth, processor time, memory, and data storage.

**T.Hack\_Comm\_Eavesdrop:** Hacker eavesdrops on user data communications  
Hacker obtains user data by eavesdropping on communications lines.

**T.Hack\_Crypto:** Cryptoanalysis for theft of information  
A hacker performs cryptoanalysis on encrypted data in order to recover message content.

**T.Hack\_Msg\_Data:** Message content modification  
A hacker modifies information intercepted from a communication link between two unsuspecting entities before passing it on, thereby deceiving the intended recipient.

**T.Hack\_Phys:** Exploitation of vulnerabilities in the physical environment of the system  
A hacker physically interacts with the system to exploit vulnerabilities in the physical environment, resulting in arbitrary security compromises.

**T.Hack\_Social\_Engineer:** Social engineering  
A hacker uses social engineering techniques to gain information about system entry, system use, system design, or system operation.

**T.IdenClone:** Identity cloning  
Identities are unique keys that must remain protected by the TPM. Creating a copy of the identity breaks the uniqueness promise.

**T.IdenPKI:** Identity PKI  
The identity creation process requires a PKI to certify the identity. This PKI must ensure the uniqueness of the identity and validate the endorsement key. Failure to properly perform these operations results in a bad identity.

**T.Malicious\_Code:** Malicious code exploitation  
An authorized user, IT system, or hacker downloads and executes malicious code, which causes abnormal processes that violate the integrity, availability, or confidentiality of system assets.

**T.MeasureFalse:** False integrity measurement  
The entity or process providing the integrity measurement provides a false value.

**T.OwnerMasquerade:** Owner masquerade  
An attacker can masquerade as the owner of either the TPM or an entity if they obtain the owner authorization data.

**T.Power\_Disrupt:** Unexpected disruption of system or component power  
A human or environmental agent disrupts power causing the system to lose information or security protection.

**T.ProtStorAttribute:** Protected storage attribute  
Each protected storage object has attributes that indicate its migration status, object type and source. Modification of these attributes allows the attacker to use the object in an unauthorized manner.

**T.ProtStoreBackup:** Protected storage backup  
The protected storage backup mechanism must provide assurances that the migration and non-migration bits are properly followed. If they are not followed then non-migratable information may move from one system to another.

**T.ProtStoreCrypto:** Protected Storage Cryptography  
Protected storage requires cryptography to protect the contents when the data is not inside the TPM. Failure of the cryptography exposes the data.

**T.ProtStoreMaintenance:** Protected storage maintenance  
The protected storage maintenance feature allows for the cloning of a TPM. If this mechanism is abused then the attacker can make copies of TPM devices.

**T.Repudiate\_Receive:** Recipient denies receiving information  
The recipient of a message denies receiving the message, to avoid accountability for receiving the message or to avoid obligations incurred as a result of receiving the message.

**T.Repudiate\_Send:** Sender denies sending information  
The sender of a message denies sending the message to avoid accountability for sending the message or to avoid obligations incurred as a result of sending the message.

**T.Repudiate\_Transact:** A participant denies performing a transaction  
A participant in a transaction denies participation in the transaction to avoid accountability for the transaction or for resulting obligations.

**T.SpecRef:** Failure to follow specification  
The developer creating the TOE does not follow the TCPA specification and makes mistakes in implementation. This creates holes in the TOE that expose user and internal information.

**T.Spoofing:** Legitimate system services are spoofed  
An attacker tricks users into interacting with spurious system services.

**T.User\_Abuse\_Conf:** Hostile user acts cause confidentiality breaches  
A user collects sensitive or proprietary information and removes it from the system.

T.User\_Collect: User abuses authorization to collect data  
User abuses granted authorizations to improperly collect sensitive or security-critical data.

T.User\_Err\_Conf: User errors cause confidentiality breaches  
A user commits errors that cause information to be delivered to the wrong place or wrong person.

T.User\_Err\_Inaccess: User error makes data inaccessible  
A user accidentally deletes user data or changes system data rendering user data inaccessible.

T.User\_Err\_Integrity: User errors cause integrity breaches  
A user commits errors that induce erroneous actions by the system and/or erroneous statements its users.

T.User\_Err\_Slf\_Protect: User errors undermine the system's security features  
A user commits errors that cause the system or one of its applications to undermine the system's security features.

T.User\_Misuse\_Avl\_Resc: User's misuse causes denial of service  
A user's unauthorized use of resources causes an undue burden on an affected resource.

T.User\_Modify: User abuses authorization to modify data  
A user abuses granted authorizations to improperly change or destroy sensitive or security-critical data.

T.User\_Send: User abuses authorization to send data  
A user abuses granted authorizations to improperly send sensitive or security-critical data.

### **3.3 - Organisational Security Policies**

P.Accountability: Individual accountability  
Individuals shall be held accountable for their actions.

P.Authorities: Notification of threats and vulnerabilities  
Appropriate authorities shall be immediately notified of any threats or vulnerabilities impacting systems that process their data.

P.Availability: Information availability  
Information shall be available to satisfy mission requirements.

P.EMI EMC: EMI Emissions  
The TOE security policy must specify what level of emissions are permissible when the TOE executes cryptographic operations.



P.Guidance: Installation and usage guidance

Guidance shall be provided for the secure installation and use of the system.

P.Information\_AC: Information access control

Information shall be accessed only by authorized individuals and processes.

P.Integrity: Information content integrity

Information shall retain its content integrity.

P.Lifecycle: System lifecycle phases integrate security

Information systems security shall be an integral part of all system lifecycle phases.

P.Marking: Information marking

Information shall be appropriately marked and labeled.

P.MessageAuth: Message authorization

Each message to a TPM protected capability uses the authorization protocol

P.Physical\_Control: Physical protection

Information shall be physically protected to prevent unauthorized disclosure, destruction, or modification.

P.SpecRef: Specification reference

The TOE must provide all features and functions of the TCPA in a consistent manner.

P.TCPAAuthorization: TCPA Authorization

The TOE must provide the ability to participate in the authorization protocol from chapter 4.

P.TCPAIdentities: TCPA Identities

The TOE must create and manage identities.

P.TCPAOwnership: TCPA TPM and entity ownership

The TOE must provide the mechanisms to create and use the ownership protocol.

P.TCPAProtectMigrate: TCPA Protected storage migration and non migration

The TOE must provide the mechanisms to identify the tree a storage entity is in (migratable or non-migratable), ensure that the label once set never changes and manage the migration, backup and recovery of storage entities.

P.TCPARegDIR: TCPA DIR registers

The TOE must supply DIR registers.

P.TCPARegPCR: TCPA PCR registers

The TOE must provide volatile PCR registers

P.TSP: TOE Security Policy

A TOE security policy (TSP) must identify all roles, services and security relevant data items, and specify what access (if any) a user, performing a service within the context of a given role, has to each of the security-relevant data items. The policy must specify that: users agree to protect keys and data access, users agree to report loss of keys or perceived compromise to security and user agree not to collude.

## 4 - Security Objectives

Identify and define the security objectives for the TOE and its environment. Security objectives should reflect the stated intent, be suitable to counter all identified threats, and cover all identified organisational security policies and assumptions.

### 4.1 - Security Objectives for the TOE

O.AC\_Label\_Export: Object security attributes and exportation

Provide object security attributes in exported data with moderate to high effectiveness. The attributes are those associated with specific security function policies.

O.Admin\_Code\_Val: Administrative validation of executables

Validate executable objects prior to allowing execution. Validation needs to be done by someone with an expertise to recognize malicious code and the authority and means to prevent its execution.

O.Admin\_Guidance: Administrator guidance documentation

Deter administrator errors by providing adequate administrator guidance.

O.Apply\_Code\_Fixes: Apply patches to fix the code

Apply patches to fix the code when vulnerabilities in code allow unauthorized and undiscovered access.

O.Atomic\_Functions: Complete security functions or recover to previous state

Recover automatically to a consistent, secure state if a security function does not complete successfully in the presence of certain types of failures.

O.AuditLog: Audit Log

The TPS shall maintain the audit log

O.Audit\_Generation: Audit records with identity

Record in audit records: date and time of action, location of the action, and the entity responsible for the action.

**O.Change\_Control\_Users:** User notification of data content changes  
Notify users of changes to data content in order to make any adjustments to their own data.

**O.Clean\_Obj\_Recovery:** Object and data recovery free from malicious code  
Recover to a viable state after malicious code is introduced and damage occurs, removing the malicious code as part of the process.

**O.Code\_Signing:** Code signing and verification  
Check verification of signed downloaded code prior to execution. A well-known example is checking digital signatures on signed Java applets.

**O.Config\_Management:** Implement operational configuration management  
Implement a configuration management plan. Implement configuration management to assure storage integrity, identification of system connectivity (software, hardware, and firmware), and identification of system components (software, hardware, and firmware).

**O.Crypto\_AC:** Cryptographic access control policy  
Restrict user access to cryptographic IT assets in accordance with a specified user access control policy.

**O.Crypto\_Data\_Sep:** Separation of cryptographic data  
Provide complete separation between plaintext and encrypted data and between data and keys. This requires separate channels and separate storage areas. The only place any data can pass between the plaintext and encrypted data modules is in the cryptographic engine. There should be no way for plaintext keys to reach either data module and no way for data to enter the key handling module. Encrypted keys can be handled as encrypted data, but with limited user access.

**O.Crypto\_Dsgn\_Impl:** Cryptographic Design and Implementation  
Minimize or even eliminate design and implementation errors in the cryptographic modules and functions.

**O.Crypto\_Import\_Export:** Cryptographic import, export, and inter-TSF transfer  
Protect cryptographic data assets when they are being transmitted to and from the TOE, either through intervening untrusted components or directly to/from human users.

**O.Crypto\_Key\_Man:** Cryptographic Key Management  
Fully define cryptographic components, functions, and interfaces. Ensure appropriate protection for cryptographic keys throughout their lifecycle, covering generation, distribution, storage, use, and destruction.

**O.Crypto\_Modular\_Dsgn:** Cryptographic Modular Design  
Prevent errors in one part of the TOE from influencing other parts, especially cryptographic parts. To this end, noncryptographic I/O paths must be well defined and

logically independent of circuitry and processes performing key generation, manual key entry, key zeroising, and similar key-related operations.

O.Crypto\_Operation: Cryptographic function definition  
Cryptographic components, functions, and interfaces shall be fully defined.

O.Crypto\_Self\_Test: Cryptographic self test  
Provide the ability to verify that the cryptographic functions operate as designed.

O.Crypto\_Test\_Reqs: Test cryptographic functionality  
Test cryptographic operation and key management.

O.Data\_Exchange\_Conf: Enforce data exchange confidentiality  
Protect user data confidentiality when exchanging data with a remote system.

O.Data\_Export\_Control: Control user data exportation  
Impose information control policies that do not allow export of specified data and/or export to specified locations.

O.Data\_Imp\_Exp\_Control: Data import/export to/from system control  
Protect data from being sent to erroneous places and more places external to the system than allowed by the organization's security policy. Conversely the import of data into the system should be protected from illicit information or information not allowed by the organization's security policy.

O.EMSEC\_Design: Provide physical emanations security  
Design and build the system in such a way as to control the production of intelligible emanations within specified limits.

O.Export\_Control: Sanitize data objects containing hidden or unused data  
Sanitize data objects that may contain hidden data when they are exported from the TOE in order to inhibit steganographic smuggling.

O.External\_Labels: Label or mark information for external systems  
Label or mark information for external systems to prevent the exchange of inappropriate data between systems.

O.Fail\_Secure: Preservation of secure state for failures in critical components  
Preserve the secure state of the system in the event of a secure component failure.

O.Fault\_Tolerance: Provide fault tolerant operations for critical components  
Provide fault tolerant operations for critical components and continue to operate in the presence of specific failures in one or more system components.

O.General\_Integ\_Checks: Periodically check integrity  
Provide periodic integrity checks on both system and user data.

**O.Hack\_Limit\_Sessions:** Limit sessions to outside users

Limit the number of sessions available to outside users. A hacker can initiate multiple communication sessions that could cause an overload on resources, for example, half open session starts as is seen in "SYN flood" attacks.

**O.Info\_Flow\_Control:** System enforced information flow

Enforce an information flow policy whereby users are constrained from allowing access to information they control, regardless of their intent (e.g., mandatory access control). This lattice property of security attributes is commonly associated with the U.S. DoD implementations of Mandatory Access Control (MAC).

**O.Input\_Inspection:** Require inspection for absence of malicious code.

Require inspection of downloads/transfers.

**O.Integ\_Data\_Mark\_Exp:** Data marking integrity export

Ensure that data markings are included with data that is exported to another trusted product.

**O.Integ\_Sys\_Data\_Ext:** Integrity of system data transferred externally

Ensure the integrity of system data exchanged externally with another trusted product by using a protocol for data transfer that will permit error detection and correction.

This includes detecting and possibly correcting errors in data received and encoding outgoing data to make it possible for the receiver to detect and possibly correct errors. The method for detecting and correcting errors is based on some method (protocol) that is agreed upon by participating parties.

**O.Integ\_Sys\_Data\_Int:** Integrity of system data transferred internally

Ensure the integrity of system data transferred internally.

**O.Integ\_User\_Data\_Int:** Protect user data during internal transfer

Ensure the integrity of user data transferred internally within the system.

**O.Integrity\_Data/SW:** Integrity protection for user data and software

Provide integrity protection for user data and software.

**O.Integrity\_Data\_Rep:** Integrity of system data replication

Ensure that when system data replication occurs across the system the data is consistent for each replication.

**O.Integrity\_Practice:** Operational integrity system function testing

Provide system functional tests to periodically test the integrity of the hardware and code running system functions.

**O.IntelEman\_Contain:** Emanations containment

Confine system-produced intelligible emanations to within a specified limit.

O.IntelEman\_Control: Emanations control

Limit system-produced intelligible emanations to within a specified limit.

O.Lifecycle\_Security: Lifecycle security

Provide tools, techniques, and security employed during the development phase. Detect and resolve flaws during the operational phase. Provide safe destruction techniques.

O.Limit\_Actions\_Auth: Restrict actions before authentication

Restrict the actions a user may perform before the TOE verifies the identity of the user.

O.Limit\_Comm\_Sessions: Limit the number of user initiated communication sessions

Provide mechanisms to limit the number of sessions that the user can initiate, if the user initiates multiple sessions that exceed the processors ability to perform in a reliable and efficient manner. These sessions could either be communication (TCP/IP) sessions or user login sessions.

O.Maintain\_Sec\_Domain: Maintain security domain

Maintain at least one security domain for system (TOE) execution to protect the TOE from interference and tampering.

O.Malicious\_Code: Procedures for preventing malicious code

Incorporate malicious code prevention procedures and mechanisms.

O.Manage\_Res\_Sec\_Attr: Manage resource security attributes

Provide management on resource security attributes.

O.Manage\_TSF\_Data: Manage security-critical data to avoid storage space being exceeded

Manage security-critical (TSF) data to ensure that the size of the data does not exceed the space allocated for storage of the data.

O.MessageAuthentication: Message authentication

Each requestor must prove knowledge of the shared secret.

O.MetricReporting: Integrity metric reporting

The TOE must report the values in the current PCR registers. The report may be digitally signed.

O.NoBore: No BORE attacks

The TOE provides protection from Break Once Run Everywhere attacks.

O.No\_Residual\_Info: Eliminate residual information

Ensure there is no "object reuse;" i.e., ensure that there is no residual information in some information containers or system resources upon their reallocation to different users.

**O.NonRepud\_Assess\_Recd:** Non-repudiation support for received information by a nonlocal sender's TSF

Support nonrepudiation for received information by supporting remote handling of nonrepudiation evidence if needed.

**O.NonRepud\_Assess\_Sent:** Non-repudiation support for sent information by the nonlocal receiving TSF.

Support nonrepudiation for sent information by supporting remote handling of nonrepudiation evidence if needed.

**O.NonRepud\_Gen\_Recd:** Non-repudiation support for received information by the recipient's TSF

Prevent a receiving user from avoiding accountability for receiving a message by providing evidence that the user received the message.

**O.NonRepud\_Gen\_Sent:** Non-repudiation support for sent information by the sender's TSF.

Prevent a user from avoiding accountability for sending a message to a recipient at a different site by providing evidence that the user sent the message.

**O.Obj\_Attr\_Integrity:** Basic object attribute integrity

Maintain object security attributes with moderate to high accuracy (under the guidance of qualified users).

**O.Obj\_Protection:** Object domain protection

Require domain protection for objects. Specify object classes (domains), user groups, and operation classes. Use these to specify which operations may be performed on which objects by which users. Basically this controls what users can do in a given group.

**O.Prevent\_Link:** Prevent linking of multiple service use

Ensure that a user may make multiple uses of a service or resource without other specified users being able to link these uses together.

**O.Protected\_Capability:** Protected Capability and shielded location

The TOE must identify and protect capabilities as defined in the TCPA specification.

**O.Rcv\_MsgMod\_ID:** Identify message modification in messages received

The TSF recognizes changes to messages that occurred in transit, including insertion of spurious messages and deletion or replay of legitimate messages.

**O.React\_Discovered\_Atk:** React to discovered attacks

Implement automated notification or other reactions to the TSF-discovered attacks in an effort to identify attacks and to create an attack deterrent.

**O.RootMeasurement:** Measurement root of trust

The entity that provides the base for measuring integrity values is the measurement root of trust. This entity on a PC would be the boot block or something similar.

**O.RootReporting:** Reporting root of trust

The reporting root of trust is the endorsement key. This provides a singular point that all challengers can rely on.

**O.SecureManufacturing:** Secure TPM creation and certification

The TPM manufacturing process requires the creation and certification of the endorsement key. The TPM manufacturing process must perform this creation and certification in a manner that provides the assurances that the endorsement key was properly created. The process must also provide assurances that the certification of the endorsement key is done with the correct private key and that the process protects the certification key and properly protects certification process.

**O.Secure\_State:** Protect and maintain secure system state

Maintain and recover to a secure state without security compromise after system error or other interruption of system operation.

**O.Security\_Attr\_Mgt:** Manage security attributes

Manage the initialization of, values for, and allowable operations on security attributes.

**O.Security\_Data\_Mgt:** Manage security-critical data

Manage the initialization of, limits on, and allowable operations on security-critical data.

**O.Security\_Func\_Mgt:** Manage behavior of security functions

Provide management mechanisms for security mechanisms.

**O.Security\_Roles:** Security roles

Maintain security-relevant roles and the association of users with those roles.

**O.Snt\_MsgMod\_ID:** Identify message modification in messages sent

The TSF supports recognition of changes to transmitted messages that occurred in transit, including insertion of spurious messages and deletion or replay of legitimate messages.

**O.Source\_Code\_Exam:** Examine the source code for developer flaws

Examine for accidental or deliberate flaws in code made by the developer. The accidental flaws could be lack of engineering detail or bad design. Where the deliberate flaws would include building trapdoors for later entry as an example.

**O.SpecRef:** Specification reference

The TOE must provide all of the features and functions as specified in the TCPA specification.



O.Standard\_Output\_Pres: Standard presentation of output values  
Present each possible output value in a standard form.

O.Storage\_Integrity: Storage integrity  
Provide integrity for data.

O.Sys\_Assur\_HW/SW/FW: Validation of security function  
Ensure that security-relevant software, hardware, and firmware are correctly functioning through features and procedures.

O.Sys\_Backup\_Procs: System backup procedures  
Provide backup procedures to ensure that the system can be reconstructed.

O.Sys\_Backup\_Verify: Detect modifications of backup hardware, firmware, software  
Detect modifications to backup hardware, firmware, and software.

O.Sys\_Self\_Protection: Protection of system security function  
Protect the system security functions through technical features.

O.TCPAIdentities: TCPA Identities  
The TOE must provide the ability to create, manage and use identities.

O.TCPAProtectedStorage: TCPA Protective Storage  
The TOE must provide a protected storage mechanism as defined in the specification section 3.6 and chapter 6.

O.TSF\_Rcv\_Err\_ID\_Loc: Local detection of received security-critical data modified in transit  
Identification by the system (TOE) of modification of security-critical (TSF) data occurring in transit from a remote trusted site must occur.

O.TSF\_Rcv\_Err\_ID\_Rem: Remote detection of received security-critical data modified in transit  
Identification by the remote site of the modification of security-critical (TSF) data occurring in transit from the remote site must occur.

O.TSF\_Snd\_Err\_ID\_Loc: Local detection of sent security-critical data modified in transit  
Identification of modification of security-critical (TSF) data occurring in transit to a remote site by the TSF must occur.

O.TSF\_Snd\_Err\_ID\_Rem: Remote detection of sent security-critical data modified in transit.  
Identification of modification of security-critical (TSF) data occurring in transit to a remote site by the remote site must occur.

O.Tamper\_ID: Tamper detection

Provide system features that detect physical tampering of a system component, and use those features to limit security breaches.

O.Tamper\_Resistance: Tamper resistance

Prevent or resist physical tampering with specified system devices and components.

O.Trusted\_Path: Provide a trusted path

Provide a trusted path between the user and the system. Execution of a user-requested action must be made via a trusted path with the following properties:

- \* The path is logically distinct from, and cannot be confused with other communication paths (by either the user or the system).
- \* The path provides assured identification of its end points.

O.Trusted\_Recovery: Trusted recovery of security functionality

Recovery to a secure state, without security compromise, after a discontinuity of operations.

O.Trusted\_Recovery\_Doc: Documentation of untrusted data recovery

Provide trusted recovery to ensure that data cannot be lost or misplaced. Any circumstances which can cause untrusted recovery to be documented with mitigating procedures established.

O.User\_Auth\_Management: User authorization management

Manage and update user authorization and privilege data in accordance with organizational security and personnel policies.

O.User\_Conf\_Prevention: Basic confidentiality-breach prevention

Prevent unauthorized export of confidential information from the TOE with moderate effectiveness.

O.User\_Data\_Integrity: Integrity protection of stored user data

Provide appropriate integrity protection for stored user data.

O.User\_Data\_Transfer: Protection of transmitted user data

Provide the ability to have physically protected communications lines, intrusion detection for communications lines, and/or need-to-know isolation for communications lines.

O.User\_Defined\_AC: User-defined access control

Enforce an access control policy whereby users may determine who may access information they control.

O.User\_Guidance: User guidance documentation

Provide documentation for the general user.

Security Objectives:

## 4.2 - Security Objectives for the Environment

O.AuditLog: Audit Log

The TPS shall maintain the audit log

O.Audit\_Protect: Protect stored audit records

Protect audit records against unauthorized access, modification, or deletion to ensure accountability of user actions.

O.Trusted\_Path: Provide a trusted path

Provide a trusted path between the user and the system. Execution of a user-requested action must be made via a trusted path with the following properties:

- \* The path is logically distinct from, and cannot be confused with other communication paths (by either the user or the system).
- \* The path provides assured identification of its end points.

# 5 - IT Security Requirements

Include an overall summary of the functional and assurance security requirements for the TOE, and environment.

## 5.1 - TOE Security Functional Requirements

### 5.1.0.1 - Security alarms (FAU\_ARP.1)

The TSF shall take Shutdown of TOE functions, upon detection of a potential security violation.<sup>FAU\_ARP.1.1</sup>

### 5.1.0.2 - Audit data generation (FAU\_GEN.1)

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

- a) Start-up and shutdown of the audit functions;
- b) All auditable events for the minimum level of audit; and
- c) all protected capabilities per TCPA specification.<sup>FAU\_GEN.1.1</sup>

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

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

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, parent entity<sup>FAU\_GEN.1.2</sup>

### 5.1.0.3 - Selective audit (FAU\_SEL.1)

The TSF shall be able to include or exclude auditable events from the set of audited events based on the following attributes:

a) event type

b) none.<sup>FAU\_SEL.1.1</sup>

### 5.1.0.4 - Selective proof of origin (FCO\_NRO.1)

The TSF shall be able to generate evidence of origin for transmitted all authorized commands per the TCPA spec at the request of the recipientnone.<sup>FCO\_NRO.1.1</sup>

The TSF shall be able to relate the nonces of the originator of the information, and the command ordinal, security sensitive parameters as per TCPA specification of the information to which the evidence applies.<sup>FCO\_NRO.1.2</sup>

The TSF shall provide a capability to verify the evidence of origin of information to recipientnone given evidence only available when requestor properly authenticates.<sup>FCO\_NRO.1.3</sup>

### 5.1.0.5 - Enforced proof of origin (FCO\_NRO.2)

The TSF shall enforce the generation of evidence of origin for transmitted all authorized commands per the TCPA spec at all times.<sup>FCO\_NRO.2.1</sup>

The TSF shall be able to relate the nonces of the originator of the information, and the command ordinal, security sensitive parameters as per TCPA specification of the information to which the evidence applies.<sup>FCO\_NRO.2.2</sup>

The TSF shall provide a capability to verify the evidence of origin of information to recipientnone given evidence only available when requestor properly authenticates.<sup>FCO\_NRO.2.3</sup>

### 5.1.0.6 - Selective proof of receipt (FCO\_NRR.1)

The TSF shall be able to generate evidence of receipt for received all authorized commands provide an authenticated return message at the request of the originatornone.<sup>FCO\_NRR.1.1</sup>

The TSF shall be able to relate the nonces of the recipient of the information, and the command ordinal, return code, security sensitive parameters as per TCPA specification of the information to which the evidence applies.<sup>FCO\_NRR.1.2</sup>

The TSF shall provide a capability to verify the evidence of receipt of information to originatornone given only available when the command properly authenticated, failures have no ability to provide evidence.<sup>FCO\_NRR.1.3</sup>

#### 5.1.0.7 - Enforced proof of receipt (FCO\_NRR.2)

The TSF shall enforce the generation of evidence of receipt for received all authorized commands provide an authenticated return message.<sup>FCO\_NRR.2.1</sup>

The TSF shall be able to relate the nonces of the recipient of the information, and the command ordinal, return code, security sensitive parameters as per TCPA specification of the information to which the evidence applies.<sup>FCO\_NRR.2.2</sup>

The TSF shall provide a capability to verify the evidence of receipt of information to originator none given only available when the command properly authenticated, failures have no ability to provide evidence.<sup>FCO\_NRR.2.3</sup>

#### 5.1.0.8 - Cryptographic key generation (FCS\_CKM.1)

The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm P1363 and specified cryptographic key sizes RSA 512, 768, 1024, 2048 that meet the following: P1363, PKCS#1 V2.<sup>FCS\_CKM.1.1</sup>

#### 5.1.0.9 - Cryptographic key destruction (FCS\_CKM.4)

The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method erasure of all memory areas that meets the following: P1363.<sup>FCS\_CKM.4.1</sup>

#### 5.1.0.10 - Cryptographic operation (FCS\_COP.1)

The TSF shall perform RSA encrypt decrypt, SHA, HMAC in accordance with a specified cryptographic algorithm RSA, SHA, HMAC and cryptographic key sizes RSA 512, 768, 1024, 2048 that meet the following: P1363, PKCS#1 V2, RFC 2104, SHA.<sup>FCS\_COP.1.1</sup>

#### 5.1.0.11 - Complete access control (FDP\_ACC.2)

The TSF shall enforce the Owner access control on TPM owner, all protected storage nodes, all identities and all operations among subjects and objects covered by the SFP.<sup>FDP\_ACC.2.1</sup>

The TSF shall ensure that all operations between any subject in the TSC and any object within the TSC are covered by an access control SFP.<sup>FDP\_ACC.2.2</sup>

#### 5.1.0.12 - Security attribute based access control (FDP\_ACF.1)

The TSF shall enforce the TCPA ownership protocol to objects based on ownership token, protected storage attributes.<sup>FDP\_ACF.1.1</sup>

The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: i) entity use requires use authorization  
ii) entity loading requires authorization of entity owner.<sup>FDP\_ACF.1.2</sup>

The TSF shall explicitly authorise access of subjects to objects based on the following

additional rules: knowledge of ownership token authorizes access to entity.<sup>FDP\_ACF.1.3</sup>  
 The TSF shall explicitly deny access of subjects to objects based on the failure to provide ownership token.<sup>FDP\_ACF.1.4</sup>

#### 5.1.0.13 - Basic data authentication (FDP\_DAU.1)

The TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of BIND, SEAL and WRAP entities, endorsement and SRK keys, identities,  
<sup>FDP\_DAU.1.1</sup>

The TSF shall provide TPM with the ability to verify evidence of the validity of the indicated information.<sup>FDP\_DAU.1.2</sup>

#### 5.1.0.14 - Export of user data with security attributes (FDP\_ETC.2)

The TSF shall enforce the TCPA ownership protocol when exporting user data, controlled under the SFP(s), outside of the TSC.<sup>FDP\_ETC.2.1</sup>

The TSF shall export the user data with the user data's associated security attributes.<sup>FDP\_ETC.2.2</sup>

The TSF shall ensure that the security attributes, when exported outside the TSC, are unambiguously associated with the exported user data.<sup>FDP\_ETC.2.3</sup>

The TSF shall enforce the following rules when user data is exported from the TSC: none.<sup>FDP\_ETC.2.4</sup>

#### 5.1.0.15 - Complete information flow control (FDP\_IFC.2)

The TSF shall enforce the TCPA flow control  
 on TCPA flow control controls

i) Endorsement key

ii) SRK

iii) protected storage nonces

iv) user data and all operations that cause that information to flow to and from subjects covered by the SFP.<sup>FDP\_IFC.2.1</sup>

The TSF shall ensure that all operations that cause any information in the TSC to flow to and from any subject in the TSC are covered by an information flow control SFP.<sup>FDP\_IFC.2.2</sup>

#### 5.1.0.16 - Simple security attributes (FDP\_IFF.1)

The TSF shall enforce the TCPA flow control based on the following types of subject and information security attributes: i) creation location (on or off TPM)

ii) migration status

iii) load location (on or off TPM).<sup>FDP\_IFF.1.1</sup>

The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: migration status must match, location must match.<sup>FDP\_IFF.1.2</sup>

The TSF shall enforce the none.<sup>FDP\_IFF.1.3</sup>

The TSF shall provide the following none.<sup>FDP\_IFF.1.4</sup>

The TSF shall explicitly authorise an information flow based on the following rules:

migration status match, TCPA backup, TCPA maintenance.<sup>FDP\_IFF.1.5</sup>

The TSF shall explicitly deny an information flow based on the following rules: i)

mismatch of migration

ii) mismatch of load location.<sup>FDP\_IFF.1.6</sup>

#### 5.1.0.17 - Import of user data without security attributes (FDP\_ITC.1)

The TSF shall enforce the TCPA ownership protocol when importing user data, controlled under the SFP, from outside of the TSC.<sup>FDP\_ITC.1.1</sup>

The TSF shall ignore any security attributes associated with the user data when imported from outside the TSC.<sup>FDP\_ITC.1.2</sup>

The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TSC: none.<sup>FDP\_ITC.1.3</sup>

#### 5.1.0.18 - Import of user data with security attributes (FDP\_ITC.2)

The TSF shall enforce the TCPA ownership protocol when importing user data, controlled under the SFP, from outside of the TSC.<sup>FDP\_ITC.2.1</sup>

The TSF shall use the security attributes associated with the imported user data.<sup>FDP\_ITC.2.2</sup>

The TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the user data received.<sup>FDP\_ITC.2.3</sup>

The TSF shall ensure that interpretation of the security attributes of the imported user data is as intended by the source of the user data.<sup>FDP\_ITC.2.4</sup>

The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TSC: none.<sup>FDP\_ITC.2.5</sup>

#### 5.1.0.19 - Transmission separation by attribute (FDP\_ITT.2)

The TSF shall enforce the TCPA ownership protocol to prevent the disclosure modification of user data when it is transmitted between physically-separated parts of the TOE.<sup>FDP\_ITT.2.1</sup>

The TSF shall separate data controlled by the SFP(s) when transmitted between physically-separated parts of the TOE, based on the values of the following: migration and end location.<sup>FDP\_ITT.2.2</sup>

#### 5.1.0.20 - Attribute-based integrity monitoring (FDP\_ITT.4)

The TSF shall enforce the TCPA ownership protocol to monitor user data transmitted between physically-separated parts of the TOE for the following errors: changes to wrapped entity, based on the following attributes: migration attribute.<sup>FDP\_ITT.4.1</sup>

Upon detection of a data integrity error, the TSF shall generate audit event, stop usage of object.<sup>FDP\_ITT.4.2</sup>

#### 5.1.0.21 - Subset residual information protection (FDP\_RIP.1)

The TSF shall ensure that any previous information content of a resource is made unavailable upon the deallocation of the resource from the following objects: all loaded keys.<sup>FDP\_RIP.1.1</sup>

#### 5.1.0.22 - Full residual information protection (FDP\_RIP.2)

The TSF shall ensure that any previous information content of a resource is made unavailable upon the deallocation of the resource from all objects.<sup>FDP\_RIP.2.1</sup>

#### 5.1.0.23 - Basic rollback (FDP\_ROL.1)

The TSF shall enforce TCPA security policy to permit the rollback of the entity loading, entity wrapping, on the entities.<sup>FDP\_ROL.1.1</sup>

The TSF shall permit operations to be rolled back within the before release of the entity to operation by other TPM functions.<sup>FDP\_ROL.1.2</sup>

#### 5.1.0.24 - Stored data integrity monitoring and action (FDP\_SDI.2)

The TSF shall monitor user data stored within the TSC for integrity errors, migration status, TPM assignment on all objects, based on the following attributes: user data attributes.<sup>FDP\_SDI.2.1</sup>

Upon detection of a data integrity error, the TSF shall rejection of load operation.<sup>FDP\_SDI.2.2</sup>

#### 5.1.0.25 - Basic data exchange confidentiality (FDP\_UCT.1)

The TSF shall enforce the TCPA ownership protocol and TCPA flow control to be able to transmit objects in a manner protected from unauthorised disclosure.<sup>FDP\_UCT.1.1</sup>

#### 5.1.0.26 - Data exchange integrity (FDP\_UIT.1)

The TSF shall enforce the TCPA flow control to be able to transmit user data in a manner protected from modification deletion insertion errors.<sup>FDP\_UIT.1.1</sup>

The TSF shall be able to determine on receipt of user data, whether modification deletion insertion has occurred.<sup>FDP\_UIT.1.2</sup>

#### 5.1.0.27 - Authentication failure handling (FIA\_AFL.1)

The TSF shall detect when 5 unsuccessful authentication attempts occur related to TPM owner, entity owner.<sup>FIA\_AFL.1.1</sup>

When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF shall rejection of session, creation of audit event.<sup>FIA\_AFL.1.2</sup>

#### 5.1.0.28 - User authentication before any action (FIA\_UAU.2)



The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.<sup>FIA\_UAU.2.1</sup>

#### 5.1.0.29 - Single-use authentication mechanisms (FIA\_UAU.4)

The TSF shall prevent reuse of authentication data related to ownership protocol from TCPA specification.<sup>FIA\_UAU.4.1</sup>

Application Note:

The ownership token from the TCPA specification is the 160 bit blob.

#### 5.1.0.30 - User identification before any action (FIA\_UID.2)

The TSF shall require each user to identify itself before allowing any other TSF-mediated actions on behalf of that user.<sup>FIA\_UID.2.1</sup>

#### 5.1.0.31 - Management of security functions behaviour (FMT\_MOF.1)

The TSF shall restrict the ability to disable enable the functions create, delete, load and change owner to owner of entity.<sup>FMT\_MOF.1.1</sup>

#### 5.1.0.32 - Management of security attributes (FMT\_MSA.1)

The TSF shall enforce the TCPA ownership protocol for TPM owner or entity owner to restrict the ability to change default query modify delete none the security attributes migration, location to TPM owner, entity owner.<sup>FMT\_MSA.1.1</sup>

#### 5.1.0.33 - Secure security attributes (FMT\_MSA.2)

The TSF shall ensure that only secure values are accepted for security attributes.<sup>FMT\_MSA.2.1</sup>

#### 5.1.0.34 - Static attribute initialisation (FMT\_MSA.3)

The TSF shall enforce the TCPA flow control to provide restrictive default values for security attributes that are used to enforce the SFP.<sup>FMT\_MSA.3.1</sup>

The TSF shall allow the TPM owner, entity owner to specify alternative initial values to override the default values when an object or information is created.<sup>FMT\_MSA.3.2</sup>

#### 5.1.0.35 - Management of TSF data (FMT\_MTD.1)

The TSF shall restrict the ability to change default query modify delete clear migration, backup the all shielded locations to owner.<sup>FMT\_MTD.1.1</sup>

#### 5.1.0.36 - Management of limits on TSF data (FMT\_MTD.2)

The TSF shall restrict the specification of the limits for endorsement key, SRK, identities to TPM owner.<sup>FMT\_MTD.2.1</sup>

The TSF shall take the following actions, if the TSF data are at, or exceed, the indicated limits: not allow additional entities.<sup>FMT\_MTD.2.2</sup>

#### 5.1.0.37 - Secure TSF data (FMT\_MTD.3)

The TSF shall ensure that only secure values are accepted for TSF data.<sup>FMT\_MTD.3.1</sup>

#### 5.1.0.38 - Revocation (FMT\_REV.1)

The TSF shall restrict the ability to revoke security attributes associated with the objects other additional resources within the TSC to TPM owner.<sup>FMT\_REV.1.1</sup>

The TSF shall enforce the rules TPM owner only.<sup>FMT\_REV.1.2</sup>

#### 5.1.0.39 - Restrictions on security roles (FMT\_SMR.2)

The TSF shall maintain the roles: TPM owner, entity owner.<sup>FMT\_SMR.2.1</sup>

The TSF shall be able to associate users with roles.<sup>FMT\_SMR.2.2</sup>

The TSF shall ensure that the conditions presentation of ownership token proves role are satisfied.<sup>FMT\_SMR.2.3</sup>

#### 5.1.0.40 - Anonymity without soliciting information (FPR\_ANO.2)

The TSF shall ensure that challengers unable to associate identity with endorsement key audit events, identity creation.<sup>FPR\_ANO.2.1</sup>

The TSF shall provide RNG, protected storage to TPM users without soliciting any reference to the real user name.<sup>FPR\_ANO.2.2</sup>

#### 5.1.0.41 - Unlinkability (FPR\_UNL.1)

The TSF shall ensure that TPM owner, entity owners are unable to determine whether create identity were caused by the same user identity to identity.<sup>FPR\_UNL.1.1</sup>

#### 5.1.0.42 - Abstract machine testing (FPT\_AMT.1)

The TSF shall run a suite of tests during initial start-up periodically during normal operation at the request of an authorised user to demonstrate the correct operation of the security assumptions provided by the abstract machine that underlies the TSF.<sup>FPT\_AMT.1.1</sup>

#### 5.1.0.43 - Failure with preservation of secure state (FPT\_FLS.1)

The TSF shall preserve a secure state when the following types of failures occur: bad RNG values, RSA encrypt decrypt failure, SHA failure, PCR failure.<sup>FPT\_FLS.1.1</sup>

#### 5.1.0.44 - Inter-TSF detection of modification (FPT\_ITL.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: messages use HMAC to determine that message is intact.<sup>FPT\_ITI.1.1</sup>

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 ignore command if modifications are detected.<sup>FPT\_ITI.1.2</sup>

#### 5.1.0.45 - TSF data transfer separation (FPT\_ITT.2)

The TSF shall protect TSF data from disclosure when it is transmitted between separate parts of the TOE.<sup>FPT\_ITT.2.1</sup>

The TSF shall separate user data from TSF data when such data is transmitted between separate parts of the TOE.<sup>FPT\_ITT.2.2</sup>

#### 5.1.0.46 - Passive detection of physical attack (FPT\_PHP.1)

The TSF shall provide unambiguous detection of physical tampering that might compromise the TSF.<sup>FPT\_PHP.1.1</sup>

The TSF shall provide the capability to determine whether physical tampering with the TSF's devices or TSF's elements has occurred.<sup>FPT\_PHP.1.2</sup>

#### 5.1.0.47 - Resistance to physical attack (FPT\_PHP.3)

The TSF shall resist removal from platform to the TPM module and the shielded locations by responding automatically such that the TSP is not violated.<sup>FPT\_PHP.3.1</sup>

#### 5.1.0.48 - Physical Emanations Security (FPT\_PHP\_EMSEC\_Design)

Process emanation<sup>FPT\_PHP\_EMSEC\_D.1</sup>

Connection emanation<sup>FPT\_PHP\_EMSEC\_D.2</sup>

#### 5.1.0.49 - Automated recovery without undue loss (FPT\_RCV.3)

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.<sup>FPT\_RCV.3.1</sup>

For RNG failing self-test, the TSF shall ensure the return of the TOE to a secure state using automated procedures.<sup>FPT\_RCV.3.2</sup>

The functions provided by the TSF to recover from failure or service discontinuity shall ensure that the secure initial state is restored without exceeding SRK - and all protected storage underneath the SRK for loss of TSF data or objects within the TSC.<sup>FPT\_RCV.3.3</sup>

The TSF shall provide the capability to determine the objects that were or were not capable of being recovered.<sup>FPT\_RCV.3.4</sup>

#### 5.1.0.50 - Function recovery (FPT\_RCV.4)

The TSF shall ensure that Protected storage - loss of info, identities - loss of protected storage, RNG - randomness failure, loss of endorsement have the property that the SF either completes successfully, or for the indicated failure scenarios, recovers to a consistent and secure state.<sup>FPT\_RCV.4.1</sup>

#### 5.1.0.51 - Replay detection (FPT\_RPL.1)

The TSF shall detect replay for the following entities: authorization requests.<sup>FPT\_RPL.1.1</sup>

The TSF shall perform generate audit event, destroy session, disable TPM when replay is detected.<sup>FPT\_RPL.1.2</sup>

#### 5.1.0.52 - SFP domain separation (FPT\_SEP.2)

The unisolated portion of the TSF shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects.<sup>FPT\_SEP.2.1</sup>

The TSF shall enforce separation between the security domains of subjects in the TSC.<sup>FPT\_SEP.2.2</sup>

The TSF shall maintain the part of the TSF related to TCPA ownership protocol and TCPA flow control in a security domain for their own execution that protects them from interference and tampering by the remainder of the TSF and by subjects untrusted with respect to those SFPs.<sup>FPT\_SEP.2.3</sup>

#### 5.1.0.53 - Mutual trusted acknowledgement (FPT\_SSP.2)

The TSF shall acknowledge, when requested by another part of the TSF, the receipt of an unmodified TSF data transmission.<sup>FPT\_SSP.2.1</sup>

The TSF shall ensure that the relevant parts of the TSF know the correct status of transmitted data among its different parts, using acknowledgements.<sup>FPT\_SSP.2.2</sup>

#### 5.1.0.54 - Inter-TSF basic TSF data consistency (FPT\_TDC.1)

The TSF shall provide the capability to consistently interpret Endorsement key, SRK when shared between the TSF and another trusted IT product.<sup>FPT\_TDC.1.1</sup>

The TSF shall use distinguish between endorsement key and SRK when interpreting the TSF data from another trusted IT product.<sup>FPT\_TDC.1.2</sup>

#### 5.1.0.55 - Internal TSF consistency (FPT\_TRC.1)

The TSF shall ensure that TSF data is consistent when replicated between parts of the TOE.<sup>FPT\_TRC.1.1</sup>

When parts of the TOE containing replicated TSF data are disconnected, the TSF shall ensure the consistency of the replicated TSF data upon reconnection before processing any requests for use of endorsement key, use of SRK.<sup>FPT\_TRC.1.2</sup>

#### 5.1.0.56 - TSF testing (FPT\_TST.1)

The TSF shall run a suite of self tests during initial start-up periodically during normal operation at the request of the authorised user at the conditions RNG runs tests to ensure that the RNG is not in a stuck state to demonstrate the correct operation of the TSF.<sup>FPT\_TST.1.1</sup>

The TSF shall provide authorised users with the capability to verify the integrity of TSF data.<sup>FPT\_TST.1.2</sup>

The TSF shall provide authorised users with the capability to verify the integrity of stored TSF executable code.<sup>FPT\_TST.1.3</sup>

#### 5.1.0.57 - Degraded fault tolerance (FRU\_FLT.1)

The TSF shall ensure the operation of RSA encrypt, RSA decrypt, RNG, RSA key generation, entity attribute enforcement when the following failures occur: RSA algorithm failure, RNG failure.<sup>FRU\_FLT.1.1</sup>

#### 5.1.0.58 - Limited priority of service (FRU\_PRS.1)

The TSF shall assign a priority to each subject in the TSF.<sup>FRU\_PRS.1.1</sup>

The TSF shall ensure that each access to entities and ownership protocol shall be mediated on the basis of the subjects assigned priority.<sup>FRU\_PRS.1.2</sup>

#### 5.1.0.59 - Basic limitation on multiple concurrent sessions (FTA\_MCS.1)

The TSF shall restrict the maximum number of concurrent sessions that belong to the same user.<sup>FTA\_MCS.1.1</sup>

The TSF shall enforce, by default, a limit of 2 sessions per user.<sup>FTA\_MCS.1.2</sup>

#### 5.1.0.60 - Inter-TSF trusted channel (FTP\_ITC.1)

The TSF shall provide a communication channel between itself and a remote trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.<sup>FTP\_ITC.1.1</sup>

The TSF shall permit the remote trusted IT product to initiate communication via the trusted channel.<sup>FTP\_ITC.1.2</sup>

The TSF shall initiate communication via the trusted channel for assignment of ownership token.<sup>FTP\_ITC.1.3</sup>

#### 5.1.0.61 - Trusted path (FTP\_TRP.1)

The TSF shall provide a communication path between itself and remote users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from modification or disclosure.<sup>FTP\_TRP.1.1</sup>

The TSF shall permit remote users to initiate communication via the trusted path.<sup>FTP\_TRP.1.2</sup>

The TSF shall require the use of the trusted path for initial user authentication protected functions.<sup>FTP\_TRP.1.3</sup>

## 5.2 - TOE Security Assurance Requirements

**Table 5-1 Assurance Requirements: EAL(3)**

Assurance Class	Assurance Components
ACM	ACM_CAP.3 ACM_SCP.1
ADO	ADO_DEL.1 ADO_IGS.1
ADV	ADV_FSP.1 ADV_HLD.2 ADV_RCR.1 ADV_SPM.1
AGD	AGD_ADM.1 AGD_USR.1
ALC	ALC_DVS.1 ALC_LCD.1
ATE	ATE_COV.2 ATE_DPT.1 ATE_FUN.1 ATE_IND.2
AVA	AVA_MSU.1 AVA_SOF.1 AVA_VLA.1

### 5.2.1 - Configuration management (ACM)

#### 5.2.1.1 - Authorisation controls (ACM\_CAP.3)

The CM system shall provide measures such that only authorised changes are made to the configuration items.<sup>ACM\_CAP.3.10C</sup>

The reference for the TOE shall be unique to each version of the TOE.<sup>ACM\_CAP.3.1C</sup>

The developer shall provide a reference for the TOE.<sup>ACM\_CAP.3.1D</sup>

The TOE shall be labelled with its reference.<sup>ACM\_CAP.3.2C</sup>

The developer shall use a CM system.<sup>ACM\_CAP.3.2D</sup>

The CM documentation shall include a configuration list and a CM plan.<sup>ACM\_CAP.3.3C</sup>

The developer shall provide CM documentation.<sup>ACM\_CAP.3.3D</sup>

The configuration list shall describe the configuration items that comprise the TOE.<sup>ACM\_CAP.3.4C</sup>

The CM documentation shall describe the method used to uniquely identify the configuration items.<sup>ACM\_CAP.3.5C</sup>

The CM system shall uniquely identify all configuration items.<sup>ACM\_CAP.3.6C</sup>

The CM plan shall describe how the CM system is used.<sup>ACM\_CAP.3.7C</sup>

The evidence shall demonstrate that the CM system is operating in accordance with the CM plan.<sup>ACM\_CAP.3.8C</sup>

The CM documentation shall provide evidence that all configuration items have been and are being effectively maintained under the CM system.<sup>ACM\_CAP.3.9C</sup>

### 5.2.1.2 - TOE CM coverage (ACM\_SCP.1)

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, and CM documentation.<sup>ACM\_SCP.1.1C</sup>

The developer shall provide CM documentation.<sup>ACM\_SCP.1.1D</sup>

The CM documentation shall describe how configuration items are tracked by the CM system.<sup>ACM\_SCP.1.2C</sup>

## 5.2.2 - Delivery and operation (ADO)

### 5.2.2.1 - Delivery procedures (ADO\_DEL.1)

The delivery documentation shall describe all procedures that are necessary to maintain security when distributing versions of the TOE to a user's site.<sup>ADO\_DEL.1.1C</sup>

The developer shall document procedures for delivery of the TOE or parts of it to the user.<sup>ADO\_DEL.1.1D</sup>

The developer shall use the delivery procedures.<sup>ADO\_DEL.1.2D</sup>

### 5.2.2.2 - Installation, generation, and start-up procedures (ADO\_IGS.1)

The documentation shall describe the steps necessary for secure installation, generation, and start-up of the TOE.<sup>ADO\_IGS.1.1C</sup>

The developer shall document procedures necessary for the secure installation, generation, and start-up of the TOE.<sup>ADO\_IGS.1.1D</sup>

## 5.2.3 - Development (ADV)

### 5.2.3.1 - Informal functional specification (ADV\_FSP.1)

The functional specification shall describe the TSF and its external interfaces using an informal style.<sup>ADV\_FSP.1.1C</sup>

The developer shall provide a functional specification.<sup>ADV\_FSP.1.1D</sup>

The functional specification shall be internally consistent.<sup>ADV\_FSP.1.2C</sup>

The functional specification shall describe the purpose and method of use of all external TSF interfaces, providing details of effects, exceptions and error messages, as appropriate.<sup>ADV\_FSP.1.3C</sup>

The functional specification shall completely represent the TSF.<sup>ADV\_FSP.1.4C</sup>

### 5.2.3.2 - Security enforcing high-level design (ADV\_HLD.2)

The presentation of the high-level design shall be informal.<sup>ADV\_HLD.2.1C</sup>

The developer shall provide the high-level design of the TSF.<sup>ADV\_HLD.2.1D</sup>

The high-level design shall be internally consistent.<sup>ADV\_HLD.2.2C</sup>

The high-level design shall describe the structure of the TSF in terms of

subsystems.<sup>ADV\_HLD.2.3C</sup>

The high-level design shall describe the security functionality provided by each subsystem of the TSF.<sup>ADV\_HLD.2.4C</sup>

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.<sup>ADV\_HLD.2.5C</sup>

The high-level design shall identify all interfaces to the subsystems of the TSF.<sup>ADV\_HLD.2.6C</sup>

The high-level design shall identify which of the interfaces to the subsystems of the TSF are externally visible.<sup>ADV\_HLD.2.7C</sup>

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.<sup>ADV\_HLD.2.8C</sup>

The high-level design shall describe the separation of the TOE into TSP-enforcing and other subsystems.<sup>ADV\_HLD.2.9C</sup>

### 5.2.3.3 - Informal correspondence demonstration (ADV\_RCR.1)

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.<sup>ADV\_RCR.1.1C</sup>

The developer shall provide an analysis of correspondence between all adjacent pairs of TSF representations that are provided.<sup>ADV\_RCR.1.1D</sup>

### 5.2.3.4 - Informal TOE security policy model (ADV\_SPM.1)

The TSP model shall be informal.<sup>ADV\_SPM.1.1C</sup>

The developer shall provide a TSP model.<sup>ADV\_SPM.1.1D</sup>

The TSP model shall describe the rules and characteristics of all policies of the TSP that can be modeled.<sup>ADV\_SPM.1.2C</sup>

The developer shall demonstrate correspondence between the functional specification and the TSP model.<sup>ADV\_SPM.1.2D</sup>

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.<sup>ADV\_SPM.1.3C</sup>

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.<sup>ADV\_SPM.1.4C</sup>

Application Note:

The inclusion of the security policy allows the correlation of the protection profile to the specification to have a complete description. The policy allows for a single spot to include the access control and execution policies that the specification requires. This document allows the developer to better understand the reasoning and requirements that the specification and profile require.



## 5.2.4 - Guidance documents (AGD)

### 5.2.4.1 - Administrator guidance (AGD\_ADM.1)

The administrator guidance shall describe the administrative functions and interfaces available to the administrator of the TOE. <sup>AGD\_ADM.1.1C</sup>

The developer shall provide administrator guidance addressed to system administrative personnel. <sup>AGD\_ADM.1.1D</sup>

The administrator guidance shall describe how to administer the TOE in a secure manner. <sup>AGD\_ADM.1.2C</sup>

The administrator guidance shall contain warnings about functions and privileges that should be controlled in a secure processing environment. <sup>AGD\_ADM.1.3C</sup>

The administrator guidance shall describe all assumptions regarding user behaviour that are relevant to secure operation of the TOE. <sup>AGD\_ADM.1.4C</sup>

The administrator guidance shall describe all security parameters under the control of the administrator, indicating secure values as appropriate. <sup>AGD\_ADM.1.5C</sup>

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. <sup>AGD\_ADM.1.6C</sup>

The administrator guidance shall be consistent with all other documentation supplied for evaluation. <sup>AGD\_ADM.1.7C</sup>

The administrator guidance shall describe all security requirements for the IT environment that are relevant to the administrator. <sup>AGD\_ADM.1.8C</sup>

### 5.2.4.2 - User guidance (AGD\_USR.1)

The user guidance shall describe the functions and interfaces available to the non-administrative users of the TOE. <sup>AGD\_USR.1.1C</sup>

The developer shall provide user guidance. <sup>AGD\_USR.1.1D</sup>

The user guidance shall describe the use of user-accessible security functions provided by the TOE. <sup>AGD\_USR.1.2C</sup>

The user guidance shall contain warnings about user-accessible functions and privileges that should be controlled in a secure processing environment. <sup>AGD\_USR.1.3C</sup>

The user guidance shall clearly present all user responsibilities necessary for secure operation of the TOE, including those related to assumptions regarding user behaviour found in the statement of TOE security environment. <sup>AGD\_USR.1.4C</sup>

The user guidance shall be consistent with all other documentation supplied for evaluation. <sup>AGD\_USR.1.5C</sup>

The user guidance shall describe all security requirements for the IT environment that are relevant to the user. <sup>AGD\_USR.1.6C</sup>

## 5.2.5 - Life cycle support (ALC)

### 5.2.5.1 - Identification of security measures (ALC\_DVS.1)

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.<sup>ALC\_DVS.1.1C</sup>

The developer shall produce development security documentation.<sup>ALC\_DVS.1.1D</sup>

The development security documentation shall provide evidence that these security measures are followed during the development and maintenance of the TOE.<sup>ALC\_DVS.1.2C</sup>

#### 5.2.5.2 - Developer defined life-cycle model (ALC\_LCD.1)

The life-cycle definition documentation shall describe the model used to develop and maintain the TOE.<sup>ALC\_LCD.1.1C</sup>

The developer shall establish a life-cycle model to be used in the development and maintenance of the TOE.<sup>ALC\_LCD.1.1D</sup>

The life-cycle model shall provide for the necessary control over the development and maintenance of the TOE.<sup>ALC\_LCD.1.2C</sup>

The developer shall provide life-cycle definition documentation.<sup>ALC\_LCD.1.2D</sup>

### 5.2.6 - Tests (ATE)

#### 5.2.6.1 - Analysis of coverage (ATE\_COV.2)

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.<sup>ATE\_COV.2.1C</sup>

The developer shall provide an analysis of the test coverage.<sup>ATE\_COV.2.1D</sup>

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.<sup>ATE\_COV.2.2C</sup>

#### 5.2.6.2 - Testing: high-level design (ATE\_DPT.1)

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.<sup>ATE\_DPT.1.1C</sup>

The developer shall provide the analysis of the depth of testing.<sup>ATE\_DPT.1.1D</sup>

#### 5.2.6.3 - Functional testing (ATE\_FUN.1)

The test documentation shall consist of test plans, test procedure descriptions, expected test results and actual test results.<sup>ATE\_FUN.1.1C</sup>

The developer shall test the TSF and document the results.<sup>ATE\_FUN.1.1D</sup>

The test plans shall identify the security functions to be tested and describe the goal of the tests to be performed.<sup>ATE\_FUN.1.2C</sup>

The developer shall provide test documentation.<sup>ATE\_FUN.1.2D</sup>

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.<sup>ATE\_FUN.1.3C</sup>

The expected test results shall show the anticipated outputs from a successful execution of the tests.<sup>ATE\_FUN.1.4C</sup>

The test results from the developer execution of the tests shall demonstrate that each tested security function behaved as specified.<sup>ATE\_FUN.1.5C</sup>

#### **5.2.6.4 - Independent testing - sample (ATE\_IND.2)**

The TOE shall be suitable for testing.<sup>ATE\_IND.2.1C</sup>

The developer shall provide the TOE for testing.<sup>ATE\_IND.2.1D</sup>

The developer shall provide an equivalent set of resources to those that were used in the developer's functional testing of the TSF.<sup>ATE\_IND.2.2C</sup>

### **5.2.7 - Vulnerability assessment (AVA)**

#### **5.2.7.1 - Examination of guidance (AVA\_MSU.1)**

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.<sup>AVA\_MSU.1.1C</sup>

The developer shall provide guidance documentation.<sup>AVA\_MSU.1.1D</sup>

The guidance documentation shall be complete, clear, consistent and reasonable.<sup>AVA\_MSU.1.2C</sup>

The guidance documentation shall list all assumptions about the intended environment.<sup>AVA\_MSU.1.3C</sup>

The guidance documentation shall list all requirements for external security measures (including external procedural, physical and personnel controls).<sup>AVA\_MSU.1.4C</sup>

#### **5.2.7.2 - Strength of TOE security function evaluation (AVA\_SOF.1)**

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.<sup>AVA\_SOF.1.1C</sup>

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.<sup>AVA\_SOF.1.1D</sup>

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.<sup>AVA\_SOF.1.2C</sup>

#### **5.2.7.3 - Developer vulnerability analysis (AVA\_VLA.1)**

The documentation shall show, for all identified vulnerabilities, that the vulnerability cannot be exploited in the intended environment for the TOE.<sup>AVA\_VLA.1.1C</sup>

The developer shall perform and document an analysis of the TOE deliverables searching

for obvious ways in which a user can violate the TSP.<sup>AVA\_VLA.1.1D</sup>  
The developer shall document the disposition of obvious vulnerabilities.<sup>AVA\_VLA.1.2D</sup>

## 5.3 - Security Requirements for the IT Environment (Optional)

### 5.3.0.1 - Potential violation analysis (FAU\_SAA.1)

The TSF shall be able to apply a set of rules in monitoring the audited events and based upon these rules indicate a potential violation of the TSP.<sup>FAU\_SAA.1.1</sup>

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

a) Accumulation or combination of authorization failure passing a threshold value known to indicate a potential security violation;

b) RNG failure.<sup>FAU\_SAA.1.2</sup>

Application Note:

The TPS can monitor the audit log and issue commands to the TPM cause it to cease providing information. The mechanism can be as simple as writing new values to all PCR registers causing all LOADS using PCR registers to fail.

### 5.3.0.2 - Audit review (FAU\_SAR.1)

The TSF shall provide authorised users with the capability to read all audit records from the audit records.<sup>FAU\_SAR.1.1</sup>

The TSF shall provide the audit records in a manner suitable for the user to interpret the information.<sup>FAU\_SAR.1.2</sup>

Application Note:

The TPM provides the audit event in a manner that only the TPS can access. This requires that the TPS block access to the audit event port.

### 5.3.0.3 - Protected audit trail storage (FAU\_STG.1)

The TSF shall protect the stored audit records from unauthorised deletion.<sup>FAU\_STG.1.1</sup>

The TSF shall be able to detect modifications to the audit records.<sup>FAU\_STG.1.2</sup>

Application Note:

The audit log is kept by the TPS, however the TPM keeps the mechanism that allows for detection of audit log tampering. This mechanism is the keeping of each audit log event in a PCR like register and then hashing in each subsequent audit event. The log, when

tampered with, will not be able to recreate the same value that is in the audit register and hence the challenger will know of audit log tampering. The challenger may not be able to recreate the damage to the log.

## 5.4 - Security Requirements for the Non-IT Environment (Optional)

Identify the IT security requirements that are to be met by the IT environment of the TOE. If the TOE has no asserted dependencies on the IT environment, this section may be omitted.

The trusted path is the establishment of the ephemeral session from the TCPA specification. This session establishes that both endpoints are known (i.e. they both have knowledge of the authentication token) and each subsequent communication requires the proving of knowledge of the ephemeral session key.

# 6 - Rationale

Include an overall summary of rationale.

## 6.1 - Introduction and TOE Description Rationale (Optional)

Presents the rationale used in the protection profile evaluation.

## 6.2 - Security Objectives Rationale

**Table 6-1 Mapping the TOE Security Environment to Security Objectives**

Policy/Threat/Assumptions	Objectives
Security Objectives for the TOE	
A.Application_use	O.Protected_Capability
A.Configuration	O.NoBore, O.Admin_Guidance, O.User_Guidance
A.Conformance	O.SpecRef

A.Hostile_User	O.NoBore
A.Outsider_Med	O.NoBore
A.Physical	O.Protected_Capability
A.System	O.SpecRef
A.TCPAIdentityCertification	O.TCPAProtectedStorage, O.TCPAIdentities
A.TCPARootMeasurement	O.SpecRef
A.Trusted_User	O.User_Defined_AC
P.Accountability	O.Audit_Generation, O.User_Defined_AC
P.Authorities	O.Admin_Guidance
P.Availability	O.Config_Management, O.Sys_Backup_Procs, O.Sys_Backup_Verify
P.EMI EMC	O.EMSEC_Design
P.Guidance	O.Admin_Guidance, O.User_Guidance
P.Information_AC	O.TCPAProtectedStorage
P.Integrity	O.Security_Attr_Mgt, O.Security_Data_Mgt, O.Security_Func_Mgt, O.Change_Control_Users, O.Trusted_Recovery_Doc, O.Integrity_Data/SW, O.Integrity_Practice, O.Malicious_Code, O.Storage_Integrity, O.Sys_Assur_HW/SW/FW, O.Config_Management, O.Sys_Self_Protection, O.User_Data_Integrity, O.User_Defined_AC, O.User_Data_Transfer
P.Lifecycle	O.Lifecycle_Security
P.Marking	O.External_Labels
P.MessageAuth	O.SpecRef, O.MessageAuthentication
P.Physical_Control	O.Tamper_ID
P.SpecRef	O.SpecRef, O.AuditLog, O.RootMeasurement, O.RootReporting, O.SecureManufacturing

P.TCPAAuthorization	O.SpecRef, O.MetricReporting, O.TCPAProtectedStorage
P.TCPAIdentities	O.TCPAIdentities, O.TCPAProtectedStorage
P.TCPAOwnership	O.SpecRef, O.TCPAProtectedStorage
P.TCPAProtectMigrate	O.TCPAProtectedStorage
P.TCPARegDIR	O.SpecRef, O.TCPAProtectedStorage
P.TCPARegPCR	O.SpecRef, O.MetricReporting, O.TCPAProtectedStorage
P.TSP	O.SpecRef, Security Objectives
T.Admin_Err_Commit	O.Admin_Guidance, O.Crypto_Key_Man, O.Security_Attr_Mgt, O.Security_Data_Mgt, O.Security_Func_Mgt, O.Security_Roles, O.Limit_Actions_Auth
T.Admin_Err_Omit	O.Admin_Guidance, O.Crypto_Key_Man, O.User_Auth_Management
T.Admin_UserPriv	O.Prevent_Link
T.Component_Failure	O.Crypto_Key_Man, O.Crypto_Data_Sep, O.Crypto_Dsgn_Impl, O.Crypto_Modular_Dsgn, O.Crypto_Operation, O.Crypto_Self_Test, O.Crypto_Test_Reqs, O.Fail_Secure, O.Fault_Tolerance, O.Secure_State
T.Dev_Flawed_Code	O.Code_Signing, O.Integ_Sys_Data_Int, O.No_Residual_Info, O.Secure_State, O.Sys_Self_Protection, O.Integ_Sys_Data_Ext, O.Source_Code_Exam
T.EndorseExpose	O.NoBore, O.Protected_Capability, O.SpecRef, O.TCPAIdentities, O.RootReporting, O.Admin_Guidance, O.Code_Signing, O.Crypto_Data_Sep, O.Crypto_Dsgn_Impl, O.Crypto_Key_Man, O.Fault_Tolerance, O.Fail_Secure, O.Integ_Sys_Data_Ext, O.Integ_Sys_Data_Int, O.MetricReporting, O.EMSEC_Design, O.Trusted_Recovery_Doc, O.Trusted_Recovery

T.Failure_DS_Comp	O.Fault_Tolerance, O.Integrity_Data_Rep
T.GlobalSecret	O.NoBore, O.Crypto_Dsgn_Impl
T.Hack_AC	O.Trusted_Path, O.Apply_Code_Fixes, O.AuditLog
T.Hack_Avl_Resource	O.Audit_Generation, O.Hack_Limit_Sessions, O.Manage_TSF_Data, O.React_Discovered_Atk, O.Data_Imp_Exp_Control, O.AuditLog
T.Hack_Comm_Eavesdrop	O.Data_Exchange_Conf
T.Hack_Crypto	O.Crypto_Data_Sep, O.EMSEC_Design, O.IntelEman_Control, O.IntelEman_Contain, O.SpecRef, O.Protected_Capability
T.Hack_Msg_Data	O.Rcv_MsgMod_ID, O.Snt_MsgMod_ID, O.TSF_Rcv_Err_ID_Loc, O.TSF_Rcv_Err_ID_Rem, O.TSF_Snd_Err_ID_Loc, O.TSF_Snd_Err_ID_Rem, O.AuditLog
T.Hack_Phys	O.EMSEC_Design, O.Tamper_ID, O.Tamper_Resistance, O.IntelEman_Contain, O.IntelEman_Control
T.Hack_Social_Engineer	O.Admin_Guidance, O.User_Guidance
T.IdenClone	O.TCPAIdentities, O.RootReporting, O.Crypto_Data_Sep, O.Export_Control, O.External_Labels, O.Integ_User_Data_Int, O.MetricReporting
T.IdenPKI	O.TCPAIdentities, O.RootReporting, O.MetricReporting
T.Malicious_Code	O.Trusted_Path, O.Admin_Code_Val, O.Clean_Obj_Recovery, O.Code_Signing, O.General_Integ_Checks, O.Obj_Protection, O.Input_Inspection, O.AuditLog
T.MeasureFalse	O.RootMeasurement
T.OwnerMasquerade	O.SpecRef, O.TCPAIdentities, O.TCPAProtectedStorage, O.Crypto_Data_Sep, O.Crypto_Dsgn_Impl, O.User_Auth_Management, O.User_Conf_Prevention
T.Power_Disrupt	O.Atomic_Functions, O.Trusted_Recovery



T.ProtStorAttribute	O.Protected_Capability, O.TCPAProtectedStorage, O.Crypto_Data_Sep, O.Data_Exchange_Conf, O.External_Labels, O.Integ_Data_Mark_Exp, O.Integ_User_Data_Int, O.Integrity_Data/SW, O.User_Data_Integrity
T.ProtStoreBackup	O.TCPAProtectedStorage, O.TCPAIdentities, O.Crypto_Dsgn_Impl, O.Export_Control, O.External_Labels, O.Integ_Data_Mark_Exp, O.Trusted_Recovery_Doc, O.Trusted_Recovery
T.ProtStoreCrypto	O.Protected_Capability, O.TCPAProtectedStorage, O.Crypto_Data_Sep, O.Crypto_Dsgn_Impl, O.User_Data_Integrity
T.ProtStoreMaintenance	O.TCPAProtectedStorage, O.Protected_Capability, O.TCPAIdentities, O.Crypto_Dsgn_Impl, O.Data_Exchange_Conf, O.Export_Control, O.External_Labels, O.Integ_Data_Mark_Exp, O.Trusted_Recovery_Doc, O.Trusted_Recovery
T.Repudiate_Receive	O.NonRepud_Assess_Recd, O.NonRepud_Gen_Recd
T.Repudiate_Send	O.NonRepud_Assess_Sent, O.NonRepud_Gen_Sent
T.Repudiate_Transact	O.NonRepud_Assess_Recd, O.NonRepud_Assess_Sent, O.NonRepud_Gen_Recd, O.NonRepud_Gen_Sent
T.SpecRef	O.SpecRef, O.Integrity_Data/SW, O.AuditLog, O.MetricReporting, O.Fail_Secure, O.Fault_Tolerance, O.General_Integ_Checks, O.Integ_Data_Mark_Exp, O.Integ_Sys_Data_Ext, O.Integ_Sys_Data_Int, O.Integ_User_Data_Int, O.Integrity_Data_Rep, O.Lifecycle_Security, O.Integrity_Practice, O.Limit_Actions_Auth, O.Maintain_Sec_Domain, O.Malicious_Code, O.Manage_Res_Sec_Attr, O.Manage_TSF_Data, O.No_Residual_Info
T.Spoofing	O.Trusted_Path
T.User_Abuse_Conf	O.Admin_Code_Val, O.Admin_Guidance, O.Data_Export_Control, O.Export_Control, O.Standard_Output_Pres, O.AuditLog, O.SpecRef
T.User_Collect	O.Audit_Generation, O.Trusted_Path, O.User_Defined_AC, O.Data_Exchange_Conf

	O.Info_Flow_Control, O.Integ_User_Data_Int, O.No_Residual_Info, O.Security_Roles
T.User_Err_Conf	O.AC_Label_Export, O.Crypto_AC, O.Crypto_Import_Export, O.Crypto_Key_Man, O.User_Conf_Prevention, O.SpecRef, O.TCPAIdentities
T.User_Err_Inaccess	O.User_Guidance, O.Security_Attr_Mgt
T.User_Err_Integrity	O.AC_Label_Export, O.Audit_Generation, O.Crypto_Import_Export, O.Info_Flow_Control, O.User_Defined_AC, O.SpecRef
T.User_Err_Slf_Protect	O.AC_Label_Export, O.Obj_Attr_Integrity
T.User_Misuse_Avl_Resc	O.Audit_Generation, O.Manage_TSF_Data, O.Tamper_ID, O.Data_Imp_Exp_Control, O.Limit_Comm_Sessions, O.Manage_Res_Sec_Attr, O.Tamper_Resistance, O.SpecRef
T.User_Modify	O.Audit_Generation, O.Config_Management, O.General_Integ_Checks, O.Info_Flow_Control, O.Integrity_Practice, O.Security_Data_Mgt, O.Security_Roles, O.User_Defined_AC, O.Integ_Sys_Data_Int, O.Maintain_Sec_Domain, O.SpecRef
T.User_Send	O.Admin_Code_Val, O.Audit_Generation, O.Export_Control, O.Standard_Output_Pres, O.Integ_Data_Mark_Exp, O.SpecRef
Security Objectives for the Environment	
P.SpecRef	O.AuditLog
T.Admin_Err_Commit	O.Audit_Protect
T.Hack_AC	O.Trusted_Path, O.AuditLog
T.Hack_Avl_Resource	O.AuditLog
T.Hack_Msg_Data	O.AuditLog
T.Malicious_Code	O.Trusted_Path, O.AuditLog
T.SpecRef	O.AuditLog

T.Spoofing	O.Trusted_Path
T.User_Abuse_Conf	O.AuditLog
T.User_Collect	O.Trusted_Path
T.User_Modify	O.Audit_Protect

**Table 6-2 Tracing of Security Objectives to the TOE Security Environment**

Objectives	Policy/Threat/Assumptions
Security Objectives for the TOE	
O.AC_Label_Export	T.User_Err_Conf, T.User_Err_Integrity, T.User_Err_Slf_Protect
O.Admin_Code_Val	T.Malicious_Code, T.User_Abuse_Conf, T.User_Send
O.Admin_Guidance	A.Configuration, P.Authorities, P.Guidance, T.Admin_Err_Commit, T.Admin_Err_Omit, T.EndorseExpose, T.Hack_Social_Engineer, T.User_Abuse_Conf
O.Apply_Code_Fixes	T.Hack_AC
O.Atomic_Functions	T.Power_Disrupt
O.AuditLog	P.SpecRef, T.Hack_AC, T.Hack_Avl_Resource, T.Hack_Msg_Data, T.Malicious_Code, T.SpecRef, T.User_Abuse_Conf
O.Audit_Generation	P.Accountability, T.Hack_Avl_Resource, T.User_Collect, T.User_Err_Integrity, T.User_Misuse_Avl_Resc, T.User_Modify, T.User_Send
O.Change_Control_Users	P.Integrity
O.Clean_Obj_Recovery	T.Malicious_Code
O.Code_Signing	T.Dev_Flawed_Code, T.EndorseExpose, T.Malicious_Code
O.Config_Management	P.Availability, P.Integrity, T.User_Modify

O.Crypto_AC	T.User_Err_Conf
O.Crypto_Data_Sep	T.Component_Failure, T.EndorseExpose, T.Hack_Crypto, T.IdenClone, T.OwnerMasquerade, T.ProtStorAttribute, T.ProtStoreCrypto
O.Crypto_Dsgn_Impl	T.Component_Failure, T.EndorseExpose, T.GlobalSecret, T.OwnerMasquerade, T.ProtStoreBackup, T.ProtStoreCrypto, T.ProtStoreMaintenance
O.Crypto_Import_Export	T.User_Err_Conf, T.User_Err_Integrity
O.Crypto_Key_Man	T.Admin_Err_Commit, T.Admin_Err_Omit, T.Component_Failure, T.EndorseExpose, T.User_Err_Conf
O.Crypto_Modular_Dsgn	T.Component_Failure
O.Crypto_Operation	T.Component_Failure
O.Crypto_Self_Test	T.Component_Failure
O.Crypto_Test_Reqs	T.Component_Failure
O.Data_Exchange_Conf	T.Hack_Comm_Eavesdrop, T.ProtStorAttribute, T.ProtStoreMaintenance, T.User_Collect
O.Data_Export_Control	T.User_Abuse_Conf
O.Data_Imp_Exp_Control	T.Hack_Avl_Resource, T.User_Misuse_Avl_Resc
O.EMSEC_Design	P.EMI EMC, T.EndorseExpose, T.Hack_Crypto, T.Hack_Phys
O.Export_Control	T.IdenClone, T.ProtStoreBackup, T.ProtStoreMaintenance, T.User_Abuse_Conf, T.User_Send
O.External_Labels	P.Marking, T.IdenClone, T.ProtStorAttribute, T.ProtStoreBackup, T.ProtStoreMaintenance
O.Fail_Secure	T.Component_Failure, T.EndorseExpose, T.SpecRef
O.Fault_Tolerance	T.Component_Failure, T.EndorseExpose, T.Failure_DS_Comp, T.SpecRef
O.General_Integ_Checks	T.Malicious_Code, T.SpecRef, T.User_Modify

O.Hack_Limit_Sessions	T.Hack_Avl_Resource
O.Info_Flow_Control	T.User_Collect, T.User_Err_Integrity, T.User_Modify
O.Input_Inspection	T.Malicious_Code
O.Integ_Data_Mark_Exp	T.ProtStorAttribute, T.ProtStoreBackup, T.ProtStoreMaintenance, T.SpecRef, T.User_Send
O.Integ_Sys_Data_Ext	T.Dev_Flawed_Code, T.EndorseExpose, T.SpecRef
O.Integ_Sys_Data_Int	T.Dev_Flawed_Code, T.EndorseExpose, T.SpecRef, T.User_Modify
O.Integ_User_Data_Int	T.IdenClone, T.ProtStorAttribute, T.SpecRef, T.User_Collect
O.Integrity_Data/SW	P.Integrity, T.ProtStorAttribute, T.SpecRef
O.Integrity_Data_Rep	T.Failure_DS_Comp, T.SpecRef
O.Integrity_Practice	P.Integrity, T.SpecRef, T.User_Modify
O.IntelEman_Contain	T.Hack_Crypto, T.Hack_Phys
O.IntelEman_Control	T.Hack_Crypto, T.Hack_Phys
O.Lifecycle_Security	P.Lifecycle, T.SpecRef
O.Limit_Actions_Auth	T.Admin_Err_Commit, T.SpecRef
O.Limit_Comm_Sessions	T.User_Misuse_Avl_Resc
O.Maintain_Sec_Domain	T.SpecRef, T.User_Modify
O.Malicious_Code	P.Integrity, T.SpecRef
O.Manage_Res_Sec_Attr	T.SpecRef, T.User_Misuse_Avl_Resc
O.Manage_TSF_Data	T.Hack_Avl_Resource, T.SpecRef, T.User_Misuse_Avl_Resc
O.MessageAuthentication	P.MessageAuth
O.MetricReporting	P.TCPAAuthorization, P.TCPARegPCR, T.EndorseExpose, T.IdenClone, T.IdenPKI, T.SpecRef

O.NoBore	A.Configuration, A.Hostile_User, A.Outsider_Med, T.EndorseExpose, T.GlobalSecret
O.No_Residual_Info	T.Dev_Flawed_Code, T.SpecRef, T.User_Collect
O.NonRepud_Assess_Recd	T.Repudiate_Receive, T.Repudiate_Transact
O.NonRepud_Assess_Sent	T.Repudiate_Send, T.Repudiate_Transact
O.NonRepud_Gen_Recd	T.Repudiate_Receive, T.Repudiate_Transact
O.NonRepud_Gen_Sent	T.Repudiate_Send, T.Repudiate_Transact
O.Obj_Attr_Integrity	T.User_Err_Slf_Protect
O.Obj_Protection	T.Malicious_Code
O.Prevent_Link	T.Admin_UserPriv
O.Protected_Capability	A.Application_use, A.Physical, T.EndorseExpose, T.Hack_Crypto, T.ProtStorAttribute, T.ProtStoreCrypto, T.ProtStoreMaintenance
O.Rcv_MsgMod_ID	T.Hack_Msg_Data
O.React_Discovered_Atk	T.Hack_Avl_Resource
O.RootMeasurement	P.SpecRef, T.MeasureFalse
O.RootReporting	P.SpecRef, T.EndorseExpose, T.IdenClone, T.IdenPKI
O.SecureManufacturing	P.SpecRef
O.Secure_State	T.Component_Failure, T.Dev_Flawed_Code
O.Security_Attr_Mgt	P.Integrity, T.Admin_Err_Commit, T.User_Err_Inaccess
O.Security_Data_Mgt	P.Integrity, T.Admin_Err_Commit, T.User_Modify
O.Security_Func_Mgt	P.Integrity, T.Admin_Err_Commit
O.Security_Roles	T.Admin_Err_Commit, T.User_Collect, T.User_Modify
O.Snt_MsgMod_ID	T.Hack_Msg_Data
O.Source_Code_Exam	T.Dev_Flawed_Code

O.SpecRef	A.Conformance, A.System, A.TCPARootMeasurement, P.MessageAuth, P.SpecRef, P.TCPAAuthorization, P.TCPAOwnership, P.TCPARegDIR, P.TCPARegPCR, P.TSP, T.EndorseExpose, T.Hack_Crypto, T.OwnerMasquerade, T.SpecRef, T.User_Abuse_Conf, T.User_Err_Conf, T.User_Err_Integrity, T.User_Misuse_Avl_Resc, T.User_Modify, T.User_Send
O.Standard_Output_Pres	T.User_Abuse_Conf, T.User_Send
O.Storage_Integrity	P.Integrity
O.Sys_Assur_HW/SW/FW	P.Integrity
O.Sys_Backup_Procs	P.Availability
O.Sys_Backup_Verify	P.Availability
O.Sys_Self_Protection	P.Integrity, T.Dev_Flawed_Code
O.TCPAIdentities	A.TCPAIdentityCertification, P.TCPAIdentities, T.EndorseExpose, T.IdenClone, T.IdenPKI, T.OwnerMasquerade, T.ProtStoreBackup, T.ProtStoreMaintenance, T.User_Err_Conf
O.TCPAProtectedStorage	A.TCPAIdentityCertification, P.Information_AC, P.TCPAAuthorization, P.TCPAIdentities, P.TCPAOwnership, P.TCPAProtectMigrate, P.TCPARegDIR, P.TCPARegPCR, T.OwnerMasquerade, T.ProtStorAttribute, T.ProtStoreBackup, T.ProtStoreCrypto, T.ProtStoreMaintenance
O.TSF_Rcv_Err_ID_Loc	T.Hack_Msg_Data
O.TSF_Rcv_Err_ID_Rem	T.Hack_Msg_Data
O.TSF_Snd_Err_ID_Loc	T.Hack_Msg_Data
O.TSF_Snd_Err_ID_Rem	T.Hack_Msg_Data
O.Tamper_ID	P.Physical_Control, T.Hack_Phys, T.User_Misuse_Avl_Resc
O.Tamper_Resistance	T.Hack_Phys, T.User_Misuse_Avl_Resc
O.Trusted_Path	T.Hack_AC, T.Malicious_Code, T.Spoofing, T.User_Collect

O.Trusted_Recovery	T.EndorseExpose, T.Power_Disrupt, T.ProtStoreBackup, T.ProtStoreMaintenance
O.Trusted_Recovery_Doc	P.Integrity, T.EndorseExpose, T.ProtStoreBackup, T.ProtStoreMaintenance
O.User_Auth_Management	T.Admin_Err_Omit, T.OwnerMasquerade
O.User_Conf_Prevention	T.OwnerMasquerade, T.User_Err_Conf
O.User_Data_Integrity	P.Integrity, T.ProtStoreAttribute, T.ProtStoreCrypto
O.User_Data_Transfer	P.Integrity
O.User_Defined_AC	A.Trusted_User, P.Accountability, P.Integrity, T.User_Collect, T.User_Err_Integrity, T.User_Modify
O.User_Guidance	A.Configuration, P.Guidance, T.Hack_Social_Engineer, T.User_Err_Inaccess
Security Objectives	P.TSP
Security Objectives for the Environment	
O.AuditLog	P.SpecRef, T.Hack_AC, T.Hack_Avl_Resource, T.Hack_Msg_Data, T.Malicious_Code, T.SpecRef, T.User_Abuse_Conf
O.Audit_Protect	T.Admin_Err_Commit, T.User_Modify
O.Trusted_Path	T.Hack_AC, T.Malicious_Code, T.Spoofing, T.User_Collect

### 6.2.1 - Policies

P.Accountability: Individual accountability  
Individuals shall be held accountable for their actions.

Example detailed policy statements:

- DP.Audit\_Generation: Audit data generation with identity  
The system shall provide the capability to ensure that all audit records include enough information to determine the date and time of action, the system locale of the action, the system entity that initiated or completed the action, the resources involved, and the action involved.



This detailed policy statement is addressed by:

1. O.Audit\_Generation: Audit records with identity  
Record in audit records: date and time of action, location of the action, and the entity responsible for the action.
- DP.User\_Defined\_AC: Discretionary access control  
The system shall provide a Discretionary Access Control (DAC) function (i.e., a user can grant access authorization to other users for data they control).

This detailed policy statement is addressed by:

1. O.User\_Defined\_AC: User-defined access control  
Enforce an access control policy whereby users may determine who may access information they control.

P.Authorities: Notification of threats and vulnerabilities  
Appropriate authorities shall be immediately notified of any threats or vulnerabilities impacting systems that process their data.

In General, P.Authorities is addressed by:

1. O.Admin\_Guidance: Administrator guidance documentation  
Deter administrator errors by providing adequate administrator guidance.

P.Availability: Information availability  
Information shall be available to satisfy mission requirements.

Example detailed policy statements:

- DP.Config\_Mgt\_Plan: Implement operational configuration management  
A configuration management plan shall be implemented by the system. The system shall implement configuration management to assure storage integrity, identification of system connectivity (software, hardware, and firmware), and identification of system components (software, hardware, and firmware).  
The system shall implement strong integrity mechanisms (integrity locks, encryption).

This detailed policy statement is addressed by:

1. O.Config\_Management: Implement operational configuration management  
Implement a configuration management plan. Implement configuration management to assure storage integrity, identification of system connectivity (software, hardware, and firmware), and identification of system components (software, hardware, and firmware).

- DP.Documented\_Recovery: Documented recovery  
The system shall provide procedures and features to assure that system recovery is done in a trusted and secure manner. Any circumstances that could result in an untrusted recovery shall be documented.

This detailed policy statement is addressed by:

1. O.Trusted\_Recovery\_Doc: Documentation of untrusted data recovery  
Provide trusted recovery to ensure that data cannot be lost or misplaced.  
Any circumstances which can cause untrusted recovery to be documented with mitigating procedures established.
- DP.Malicious\_Code: Malicious code prevention  
Procedures and mechanisms to prevent the introduction of malicious code into the system shall be provided.

This detailed policy statement is addressed by:

1. O.Malicious\_Code: Procedures for preventing malicious code  
Incorporate malicious code prevention procedures and mechanisms.
- DP.Sys\_Assur\_HW/SW/FW: Validation of security function integrity  
Features and procedures to validate the integrity and the expected operation of the security-relevant software, hardware, and firmware shall be provided by the system.

This detailed policy statement is addressed by:

1. O.Sys\_Assur\_HW/SW/FW: Validation of security function  
Ensure that security-relevant software, hardware, and firmware are correctly functioning through features and procedures.
- DP.Sys\_Backup\_Procs: System backup procedures  
Provide the capability to restore the system to a secure state after discontinuities of system operations.

This detailed policy statement is addressed by:

1. O.Sys\_Backup\_Procs: System backup procedures  
Provide backup procedures to ensure that the system can be reconstructed.
- DP.Sys\_Backup\_Verify: Backup protection and restoration  
The system shall provide appropriate physical and technical protection of the backup and restoration hardware, firmware, and software.

Safeguard Application: The objective O.Sys\_Backup\_Verify does not provide complete coverage of this policy. Additional objectives need to be defined. Currently detection of failure is accounted for but not prevention of failure.

This detailed policy statement is addressed by:

1. O.Sys\_Backup\_Verify: Detect modifications of backup hardware, firmware, software  
Detect modifications to backup hardware, firmware, and software.
- DP.System\_Recovery: Trusted system recovery  
Provide procedures and features to assure that system recovery is done in a trusted and secure manner.

This detailed policy statement is addressed by:

1. O.Trusted\_Recovery\_Doc: Documentation of untrusted data recovery  
Provide trusted recovery to ensure that data cannot be lost or misplaced.  
Any circumstances which can cause untrusted recovery to be documented with mitigating procedures established.
- DP.User\_Data\_Storage: Protection of stored user data  
The system shall provide appropriate storage, continuous personnel access control storage, or encrypted storage of data based on the sensitivity of the data.

Safeguard Application: O.User\_Data\_Integrity covers part of this policy, but an additional objective dealing with confidentiality may be needed.

This detailed policy statement is addressed by:

1. O.User\_Data\_Integrity: Integrity protection of stored user data  
Provide appropriate integrity protection for stored user data.
  2. O.User\_Defined\_AC: User-defined access control  
Enforce an access control policy whereby users may determine who may access information they control.
- DP.User\_Data\_Transfer: Protection of transmitted user data  
The system shall provide a protected distribution system for data transmitted.

This detailed policy statement is addressed by:

1. O.User\_Data\_Transfer: Protection of transmitted user data  
Provide the ability to have physically protected communications lines, intrusion detection for communications lines, and/or need-to-know isolation for communications lines.

**P.EMI EMC: EMI Emissions**

The TOE security policy must specify what level of emissions are permissible when the TOE executes cryptographic operations.

In General, P.EMI EMC is addressed by:

1. O.EMSEC\_Design: Provide physical emanations security  
Design and build the system in such a way as to control the production of intelligible emanations within specified limits.

P.Guidance: Installation and usage guidance

Guidance shall be provided for the secure installation and use of the system.

In General, P.Guidance is addressed by:

1. O.Admin\_Guidance: Administrator guidance documentation  
Deter administrator errors by providing adequate administrator guidance.
2. O.User\_Guidance: User guidance documentation  
Provide documentation for the general user.

P.Information\_AC: Information access control

Information shall be accessed only by authorized individuals and processes.

Example detailed policy statements:

- DP.Admin\_Security\_Data: Changes to security data by authorized personnel  
Provide mechanisms to assure that changes to security related data are executed only by authorized personnel.

This detailed policy statement is addressed by:

1. O.Security\_Attr\_Mgt: Manage security attributes  
Manage the initialization of, values for, and allowable operations on security attributes.
2. O.Security\_Data\_Mgt: Manage security-critical data  
Manage the initialization of, limits on, and allowable operations on security-critical data.
3. O.Security\_Func\_Mgt: Manage behavior of security functions  
Provide management mechanisms for security mechanisms.

- DP.User\_Defined\_AC: Discretionary access control  
The system shall provide a Discretionary Access Control (DAC) function (i.e., a user can grant access authorization to other users for data they control).

This detailed policy statement is addressed by:

1. O.User\_Defined\_AC: User-defined access control  
Enforce an access control policy whereby users may determine who may access information they control.

In General, P.Information\_AC is addressed by:

1. O.TCPAProtectedStorage: TCPA Protective Storage  
The TOE must provide a protected storage mechanism as defined in the specification section 3.6 and chapter 6.

P.Integrity: Information content integrity  
Information shall retain its content integrity.

Example detailed policy statements:

- DP.Admin\_Security\_Data: Changes to security data by authorized personnel  
Provide mechanisms to assure that changes to security related data are executed only by authorized personnel.

This detailed policy statement is addressed by:

1. O.Security\_Attr\_Mgt: Manage security attributes  
Manage the initialization of, values for, and allowable operations on security attributes.
  2. O.Security\_Data\_Mgt: Manage security-critical data  
Manage the initialization of, limits on, and allowable operations on security-critical data.
  3. O.Security\_Func\_Mgt: Manage behavior of security functions  
Provide management mechanisms for security mechanisms.
- DP.Change\_Control\_Users: Notification of data content changes  
Notify user of the time and date of the last modification of data.

This detailed policy statement is addressed by:

1. O.Change\_Control\_Users: User notification of data content changes  
Notify users of changes to data content in order to make any adjustments to their own data.
- DP.Config\_Mgt\_Plan: Implement operational configuration management  
A configuration management plan shall be implemented by the system. The system shall implement configuration management to assure storage integrity, identification of system connectivity (software, hardware, and firmware), and identification of system components (software, hardware, and firmware).  
The system shall implement strong integrity mechanisms (integrity locks, encryption).

This detailed policy statement is addressed by:

1. O.Config\_Management: Implement operational configuration management  
Implement a configuration management plan. Implement configuration management to assure storage integrity, identification of system connectivity (software, hardware, and firmware), and identification of system components (software, hardware, and firmware).

- DP.Documented\_Recovery: Documented recovery  
The system shall provide procedures and features to assure that system recovery is done in a trusted and secure manner. Any circumstances that could result in an untrusted recovery shall be documented.

This detailed policy statement is addressed by:

1. O.Trusted\_Recovery\_Doc: Documentation of untrusted data recovery  
Provide trusted recovery to ensure that data cannot be lost or misplaced.  
Any circumstances which can cause untrusted recovery to be documented with mitigating procedures established.

- DP.Integrity\_Data/SW: Strong integrity mechanisms  
The system shall implement strong integrity mechanisms (integrity locks, encryption).

This detailed policy statement is addressed by:

1. O.Integrity\_Data/SW: Integrity protection for user data and software  
Provide integrity protection for user data and software.

- DP.Integrity\_Practice: Operational integrity system function testing  
Provide system functional tests to periodically test the integrity of the hardware and code running system functions.

This detailed policy statement is addressed by:

1. O.Integrity\_Practice: Operational integrity system function testing  
Provide system functional tests to periodically test the integrity of the hardware and code running system functions.

- DP.Malicious\_Code: Malicious code prevention  
Procedures and mechanisms to prevent the introduction of malicious code into the system shall be provided.

This detailed policy statement is addressed by:

1. O.Malicious\_Code: Procedures for preventing malicious code  
Incorporate malicious code prevention procedures and mechanisms.

- DP.Storage\_Integrity: Assurance of effective storage integrity  
The system shall provide assurance that storage integrity is effective.

This detailed policy statement is addressed by:

1. O.Storage\_Integrity: Storage integrity  
Provide integrity for data.

- DP.Sys\_Assur\_HW/SW/FW: Validation of security function integrity  
Features and procedures to validate the integrity and the expected operation of the security-relevant software, hardware, and firmware shall be provided by the system.

This detailed policy statement is addressed by:

1. O.Sys\_Assur\_HW/SW/FW: Validation of security function  
Ensure that security-relevant software, hardware, and firmware are correctly functioning through features and procedures.
- DP.System\_Protection: Protection from security function modification  
Provide features or procedures for protection of the system from improper changes.

This detailed policy statement is addressed by:

1. O.Config\_Management: Implement operational configuration management  
Implement a configuration management plan. Implement configuration management to assure storage integrity, identification of system connectivity (software, hardware, and firmware), and identification of system components (software, hardware, and firmware).
  2. O.Sys\_Self\_Protection: Protection of system security function  
Protect the system security functions through technical features.
- DP.System\_Recovery: Trusted system recovery  
Provide procedures and features to assure that system recovery is done in a trusted and secure manner.

This detailed policy statement is addressed by:

1. O.Trusted\_Recovery\_Doc: Documentation of untrusted data recovery  
Provide trusted recovery to ensure that data cannot be lost or misplaced.  
Any circumstances which can cause untrusted recovery to be documented with mitigating procedures established.
- DP.User\_Data\_Storage: Protection of stored user data  
The system shall provide appropriate storage, continuous personnel access control storage, or encrypted storage of data based on the sensitivity of the data.

Safeguard Application: O.User\_Data\_Integrity covers part of this policy, but an additional objective dealing with confidentiality may be needed.

This detailed policy statement is addressed by:

1. O.User\_Data\_Integrity: Integrity protection of stored user data  
Provide appropriate integrity protection for stored user data.
2. O.User\_Defined\_AC: User-defined access control  
Enforce an access control policy whereby users may determine who may access information they control.

- DP.User\_Data\_Transfer: Protection of transmitted user data  
The system shall provide a protected distribution system for data transmitted.

This detailed policy statement is addressed by:

1. O.User\_Data\_Transfer: Protection of transmitted user data  
Provide the ability to have physically protected communications lines, intrusion detection for communications lines, and/or need-to-know isolation for communications lines.

P.Lifecycle: System lifecycle phases integrate security  
Information systems security shall be an integral part of all system lifecycle phases.

In General, P.Lifecycle is addressed by:

1. O.Lifecycle\_Security: Lifecycle security  
Provide tools, techniques, and security employed during the development phase. Detect and resolve flaws during the operational phase. Provide safe destruction techniques.

P.Marking: Information marking  
Information shall be appropriately marked and labeled.

Example detailed policy statements:

- DP.Config\_Mgt\_Plan: Implement operational configuration management  
A configuration management plan shall be implemented by the system. The system shall implement configuration management to assure storage integrity, identification of system connectivity (software, hardware, and firmware), and identification of system components (software, hardware, and firmware).  
The system shall implement strong integrity mechanisms (integrity locks, encryption).

This detailed policy statement is addressed by:

1. O.Config\_Management: Implement operational configuration management  
Implement a configuration management plan. Implement configuration management to assure storage integrity, identification of system connectivity (software, hardware, and firmware), and identification of system components (software, hardware, and firmware).

In General, P.Marking is addressed by:

1. O.External\_Labels: Label or mark information for external systems  
Label or mark information for external systems to prevent the exchange of inappropriate data between systems.



P.MessageAuth: Message authorization

Each message to a TPM protected capability uses the authorization protocol

Coverage Rationale: TCPA specification

In General, P.MessageAuth is addressed by:

1. O.SpecRef: Specification reference  
The TOE must provide all of the features and functions as specified in the TCPA specification.
2. O.MessageAuthentication: Message authentication  
Each requestor must prove knowledge of the shared secret.

P.Physical\_Control: Physical protection

Information shall be physically protected to prevent unauthorized disclosure, destruction, or modification.

In General, P.Physical\_Control is addressed by:

1. O.Tamper\_ID: Tamper detection  
Provide system features that detect physical tampering of a system component, and use those features to limit security breaches.

P.SpecRef: Specification reference

The TOE must provide all features and functions of the TCPA in a consistent manner.

In General, P.SpecRef is addressed by:

1. O.SpecRef: Specification reference  
The TOE must provide all of the features and functions as specified in the TCPA specification.
2. O.AuditLog: Audit Log  
The TPS shall maintain the audit log
3. O.RootMeasurement: Measurement root of trust  
The entity that provides the base for measuring integrity values is the measurement root of trust. This entity on a PC would be the boot block or something similar.
4. O.RootReporting: Reporting root of trust  
The reporting root of trust is the endorsement key. This provides a singular point that all challengers can rely on.
5. O.SecureManufacturing: Secure TPM creation and certification  
The TPM manufacturing process requires the creation and certification of the endorsement key. The TPM manufacturing process must perform this creation and certification in a manner that provides the assurances that the endorsement key was properly created. The process must also provide assurances that the

certification of the endorsement key is done with the correct private key and that the process protects the certification key and properly protects certification process.

P.TCPAAuthorization: TCPA Authorization

The TOE must provide the ability to participate in the authorization protocol from chapter 4.

Coverage Rationale: Required by TCPA specification

In General, P.TCPAAuthorization is addressed by:

1. O.SpecRef: Specification reference  
The TOE must provide all of the features and functions as specified in the TCPA specification.
2. O.MetricReporting: Integrity metric reporting  
The TOE must report the values in the current PCR registers. The report may be digitally signed.
3. O.TCPAProtectedStorage: TCPA Protective Storage  
The TOE must provide a protected storage mechanism as defined in the specification section 3.6 and chapter 6.

P.TCPAIdentities: TCPA Identities

The TOE must create and manage identities.

Coverage Rationale: TCPA specification

In General, P.TCPAIdentities is addressed by:

1. O.TCPAIdentities: TCPA Identities  
The TOE must provide the ability to create, manage and use identities.
2. O.TCPAProtectedStorage: TCPA Protective Storage  
The TOE must provide a protected storage mechanism as defined in the specification section 3.6 and chapter 6.

P.TCPAOwnership: TCPA TPM and entity ownership

The TOE must provide the mechanisms to create and use the ownership protocol.

Coverage Rationale: TCPA specification

In General, P.TCPAOwnership is addressed by:

1. O.SpecRef: Specification reference  
The TOE must provide all of the features and functions as specified in the TCPA specification.

2. O.TCPAProtectedStorage: TCPA Protective Storage  
The TOE must provide a protected storage mechanism as defined in the specification section 3.6 and chapter 6.

P.TCPAProtectMigrate: TCPA Protected storage migration and non migration  
The TOE must provide the mechanisms to identify the tree a storage entity is in (migratable or non-migratable), ensure that the label once set never changes and manage the migration, backup and recovery of storage entities.

Coverage Rationale: TCPA specification

In General, P.TCPAProtectMigrate is addressed by:

1. O.TCPAProtectedStorage: TCPA Protective Storage  
The TOE must provide a protected storage mechanism as defined in the specification section 3.6 and chapter 6.

P.TCPARegDIR: TCPA DIR registers  
The TOE must supply DIR registers.

Coverage Rationale: TCPA specification

In General, P.TCPARegDIR is addressed by:

1. O.SpecRef: Specification reference  
The TOE must provide all of the features and functions as specified in the TCPA specification.
2. O.TCPAProtectedStorage: TCPA Protective Storage  
The TOE must provide a protected storage mechanism as defined in the specification section 3.6 and chapter 6.

P.TCPARegPCR: TCPA PCR registers  
The TOE must provide volatile PCR registers

Coverage Rationale: TCPA specification

In General, P.TCPARegPCR is addressed by:

1. O.SpecRef: Specification reference  
The TOE must provide all of the features and functions as specified in the TCPA specification.
2. O.MetricReporting: Integrity metric reporting  
The TOE must report the values in the current PCR registers. The report may be digitally signed.

3. O.TCPAProtectedStorage: TCPA Protective Storage  
The TOE must provide a protected storage mechanism as defined in the specification section 3.6 and chapter 6.

**P.TSP: TOE Security Policy**

A TOE security policy (TSP) must identify all roles, services and security relevant data items, and specify what access (if any) a user, performing a service within the context of a given role, has to each of the security-relevant data items. The policy must specify that: users agree to protect keys and data access, users agree to report loss of keys or perceived compromise to security and user agree not to collude.

In General, P.TSP is addressed by:

1. O.SpecRef: Specification reference  
The TOE must provide all of the features and functions as specified in the TCPA specification.
2. Security Objectives:

## **6.2.2 - Threats**

**T.Admin\_Err\_Commit: Administrative errors of commission**

An administrator commits errors that directly compromise organizational security objectives or change the technical security policy enforced by the system or application.

Example Attacks:

- DA.Adm\_Err\_Crypto: Accidental mismanagement of cryptographic functions  
An administrator misconfigures cryptographic functions or stores plaintext keys in insecure areas.

This Attack is addressed by:

1. O.Crypto\_Key\_Man: Cryptographic Key Management  
Fully define cryptographic components, functions, and interfaces. Ensure appropriate protection for cryptographic keys throughout their lifecycle, covering generation, distribution, storage, use, and destruction.
- DA.Admin\_Err\_AC\_Policy: Administrator error modifies access control or information flow policy  
An administrator's error in data entry changes the access control or information flow policy enforced by the system in such a way that it no longer serves its intended purpose.

This Attack is addressed by:

1. O.Admin\_Guidance: Administrator guidance documentation  
Deter administrator errors by providing adequate administrator guidance.

Objective Component Application: Administrator guidance shall address administrator errors that change the access control or information flow policy enforced by the system or application in such a way that it no longer serves its intended purpose.

- DA.Admin\_Err\_Audit: Administrator error changes audit behavior  
An administrator's error in data entry changes the audit behavior of the system in such a way that auditing no longer serves its intended purpose.

This Attack is addressed by:

1. O.Admin\_Guidance: Administrator guidance documentation  
Deter administrator errors by providing adequate administrator guidance.

Objective Component Application: Administrator guidance shall address errors that change the audit policy enforced by the TSF.

In General, T.Admin\_Err\_Commit is addressed by:

1. O.Audit\_Protect: Protect stored audit records  
Protect audit records against unauthorized access, modification, or deletion to ensure accountability of user actions.
2. O.Security\_Attr\_Mgt: Manage security attributes  
Manage the initialization of, values for, and allowable operations on security attributes.
3. O.Security\_Data\_Mgt: Manage security-critical data  
Manage the initialization of, limits on, and allowable operations on security-critical data.
4. O.Security\_Func\_Mgt: Manage behavior of security functions  
Provide management mechanisms for security mechanisms.
5. O.Security\_Roles: Security roles  
Maintain security-relevant roles and the association of users with those roles.
6. O.Limit\_Actions\_Auth: Restrict actions before authentication  
Restrict the actions a user may perform before the TOE verifies the identity of the user.

T.Admin\_Err\_Omit: Administrative errors of omission  
The system administrator fails to perform some function essential to security.

Example Attacks:

- DA.Adm\_Err\_Crypto: Accidental mismanagement of cryptographic functions  
An administrator misconfigures cryptographic functions or stores plaintext keys in insecure areas.

This Attack is addressed by:

1. O.Crypto\_Key\_Man: Cryptographic Key Management  
Fully define cryptographic components, functions, and interfaces. Ensure appropriate protection for cryptographic keys throughout their lifecycle, covering generation, distribution, storage, use, and destruction.

In General, T.Admin\_Err\_Omit is addressed by:

1. O.Admin\_Guidance: Administrator guidance documentation  
Deter administrator errors by providing adequate administrator guidance.
2. O.User\_Auth\_Management: User authorization management  
Manage and update user authorization and privilege data in accordance with organizational security and personnel policies.

T.Admin\_UserPriv: Administrator violates user privacy policy

An administrator learns the identity (or other privacy related information) of user(s) in violation of user privacy policy. Privacy-related information is sensitive information associated with the identity of a user.

In General, T.Admin\_UserPriv is addressed by:

1. O.Prevent\_Link: Prevent linking of multiple service use  
Ensure that a user may make multiple uses of a service or resource without other specified users being able to link these uses together.

T.Component\_Failure: A critical system component fails

Failure of one or more system components results in the loss of system-critical functionality.

In General, T.Component\_Failure is addressed by:

1. O.Crypto\_Key\_Man: Cryptographic Key Management  
Fully define cryptographic components, functions, and interfaces. Ensure appropriate protection for cryptographic keys throughout their lifecycle, covering generation, distribution, storage, use, and destruction.
2. O.Crypto\_Data\_Sep: Separation of cryptographic data  
Provide complete separation between plaintext and encrypted data and between data and keys. This requires separate channels and separate storage areas. The only place any data can pass between the plaintext and encrypted data modules is in the cryptographic engine. There should be no way for plaintext keys to reach either data module and no way for data to enter the key handling module. Encrypted keys can be handled as encrypted data, but with limited user access.
3. O.Crypto\_Dsgn\_Impl: Cryptographic Design and Implementation  
Minimize or even eliminate design and implementation errors in the cryptographic modules and functions.
4. O.Crypto\_Modular\_Dsgn: Cryptographic Modular Design  
Prevent errors in one part of the TOE from influencing other parts, especially

cryptographic parts. To this end, noncryptographic I/O paths must be well defined and logically independent of circuitry and processes performing key generation, manual key entry, key zeroising, and similar key-related operations.

5. O.Crypto\_Operation: Cryptographic function definition  
Cryptographic components, functions, and interfaces shall be fully defined.
6. O.Crypto\_Self\_Test: Cryptographic self test  
Provide the ability to verify that the cryptographic functions operate as designed.
7. O.Crypto\_Test\_Reqs: Test cryptographic functionality  
Test cryptographic operation and key management.
8. O.Fail\_Secure: Preservation of secure state for failures in critical components  
Preserve the secure state of the system in the event of a secure component failure.
9. O.Fault\_Tolerance: Provide fault tolerant operations for critical components  
Provide fault tolerant operations for critical components and continue to operate in the presence of specific failures in one or more system components.
10. O.Secure\_State: Protect and maintain secure system state  
Maintain and recover to a secure state without security compromise after system error or other interruption of system operation.

T.Dev\_Flawed\_Code: Software containing security-related flaws

A system or applications developer delivers code that does not perform according to specifications or contains security flaws.

In General, T.Dev\_Flawed\_Code is addressed by:

1. O.Code\_Signing: Code signing and verification  
Check verification of signed downloaded code prior to execution. A well-known example is checking digital signatures on signed Java applets.
2. O.Integ\_Sys\_Data\_Int: Integrity of system data transferred internally  
Ensure the integrity of system data transferred internally.
3. O.No\_Residual\_Info: Eliminate residual information  
Ensure there is no "object reuse;" i.e., ensure that there is no residual information in some information containers or system resources upon their reallocation to different users.
4. O.Secure\_State: Protect and maintain secure system state  
Maintain and recover to a secure state without security compromise after system error or other interruption of system operation.
5. O.Sys\_Self\_Protection: Protection of system security function  
Protect the system security functions through technical features.
6. O.Integ\_Sys\_Data\_Ext: Integrity of system data transferred externally  
Ensure the integrity of system data exchanged externally with another trusted product by using a protocol for data transfer that will permit error detection and correction.

This includes detecting and possibly correcting errors in data received and encoding outgoing data to make it possible for the receiver to detect and possibly

correct errors. The method for detecting and correcting errors is based on some method (protocol) that is agreed upon by participating parties.

7. O.Source\_Code\_Exam: Examine the source code for developer flaws  
Examine for accidental or deliberate flaws in code made by the developer. The accidental flaws could be lack of engineering detail or bad design. Where the deliberate flaws would include building trapdoors for later entry as an example.

T.EndorseExpose: Exposure of endorsement key

The endorsement key provides the root of reporting trust. If exposed it provides the attacker numerous mechanisms that allow for the forgery, cloning and masquerading as a valid TPM

In General, T.EndorseExpose is addressed by:

1. O.NoBore: No BORE attacks  
The TOE provides protection from Break Once Run Everywhere attacks.
2. O.Protected\_Capability: Protected Capability and shielded location  
The TOE must identify and protect capabilities as defined in the TCPA specification.
3. O.SpecRef: Specification reference  
The TOE must provide all of the features and functions as specified in the TCPA specification.
4. O.TCPAIdentities: TCPA Identities  
The TOE must provide the ability to create, manage and use identities.
5. O.RootReporting: Reporting root of trust  
The reporting root of trust is the endorsement key. This provides a singular point that all challengers can rely on.
6. O.Admin\_Guidance: Administrator guidance documentation  
Deter administrator errors by providing adequate administrator guidance.
7. O.Code\_Signing: Code signing and verification  
Check verification of signed downloaded code prior to execution. A well-known example is checking digital signatures on signed Java applets.
8. O.Crypto\_Data\_Sep: Separation of cryptographic data  
Provide complete separation between plaintext and encrypted data and between data and keys. This requires separate channels and separate storage areas. The only place any data can pass between the plaintext and encrypted data modules is in the cryptographic engine. There should be no way for plaintext keys to reach either data module and no way for data to enter the key handling module. Encrypted keys can be handled as encrypted data, but with limited user access.
9. O.Crypto\_Dsgn\_Impl: Cryptographic Design and Implementation  
Minimize or even eliminate design and implementation errors in the cryptographic modules and functions.
10. O.Crypto\_Key\_Man: Cryptographic Key Management  
Fully define cryptographic components, functions, and interfaces. Ensure appropriate protection for cryptographic keys throughout their lifecycle, covering generation, distribution, storage, use, and destruction.



11. O.Fault\_Tolerance: Provide fault tolerant operations for critical components  
Provide fault tolerant operations for critical components and continue to operate in the presence of specific failures in one or more system components.
12. O.Fail\_Secure: Preservation of secure state for failures in critical components  
Preserve the secure state of the system in the event of a secure component failure.
13. O.Integ\_Sys\_Data\_Ext: Integrity of system data transferred externally  
Ensure the integrity of system data exchanged externally with another trusted product by using a protocol for data transfer that will permit error detection and correction.

This includes detecting and possibly correcting errors in data received and encoding outgoing data to make it possible for the receiver to detect and possibly correct errors. The method for detecting and correcting errors is based on some method (protocol) that is agreed upon by participating parties.

14. O.Integ\_Sys\_Data\_Int: Integrity of system data transferred internally  
Ensure the integrity of system data transferred internally.
15. O.MetricReporting: Integrity metric reporting  
The TOE must report the values in the current PCR registers. The report may be digitally signed.
16. O.EMSEC\_Design: Provide physical emanations security  
Design and build the system in such a way as to control the production of intelligible emanations within specified limits.
17. O.Trusted\_Recovery\_Doc: Documentation of untrusted data recovery  
Provide trusted recovery to ensure that data cannot be lost or misplaced. Any circumstances which can cause untrusted recovery to be documented with mitigating procedures established.
18. O.Trusted\_Recovery: Trusted recovery of security functionality  
Recovery to a secure state, without security compromise, after a discontinuity of operations.

T.Failure\_DS\_Comp: Failure of a distributed system component

Failure of a component that is part of a distributed system will cause other parts of the distributed system to malfunction or provide unreliable results.

In General, T.Failure\_DS\_Comp is addressed by:

1. O.Fault\_Tolerance: Provide fault tolerant operations for critical components  
Provide fault tolerant operations for critical components and continue to operate in the presence of specific failures in one or more system components.
2. O.Integrity\_Data\_Rep: Integrity of system data replication  
Ensure that when system data replication occurs across the system the data is consistent for each replication.

T.GlobalSecret: Global secret exposure

If the TOE has a global secret known to all TOE's then exposure of one TOE exposes all TOE's.

Coverage Rationale: This threat creates a system then when broken breaks all systems, avoidance of this situation is normally a good design.

In General, T.GlobalSecret is addressed by:

1. O.NoBore: No BORE attacks  
The TOE provides protection from Break Once Run Everywhere attacks.
2. O.Crypto\_Dsgn\_Impl: Cryptographic Design and Implementation  
Minimize or even eliminate design and implementation errors in the cryptographic modules and functions.

T.Hack\_AC: Hacker undetected system access

A hacker gains undetected access to a system due to missing, weak and/or incorrectly implemented access control causing potential violations of integrity, confidentiality, or availability.

In General, T.Hack\_AC is addressed by:

1. O.Trusted\_Path: Provide a trusted path  
Provide a trusted path between the user and the system. Execution of a user-requested action must be made via a trusted path with the following properties:  
\* The path is logically distinct from, and cannot be confused with other communication paths (by either the user or the system).  
\* The path provides assured identification of its end points.
2. O.Apply\_Code\_Fixes: Apply patches to fix the code  
Apply patches to fix the code when vulnerabilities in code allow unauthorized and undiscovered access.
3. O.AuditLog: Audit Log  
The TPS shall maintain the audit log

T.Hack\_Avl\_Resource: Hacker attempts resource denial of service

A hacker executes commands, sends data, or performs other operations that make system resources unavailable to system users. Resources that may be denied to users include bandwidth, processor time, memory, and data storage.

In General, T.Hack\_Avl\_Resource is addressed by:

1. O.Audit\_Generation: Audit records with identity  
Record in audit records: date and time of action, location of the action, and the entity responsible for the action.
2. O.Hack\_Limit\_Sessions: Limit sessions to outside users  
Limit the number of sessions available to outside users. A hacker can initiate multiple communication sessions that could cause an overload on resources, for example, half open session starts as is seen in "SYN flood" attacks.

3. O.Manage\_TSF\_Data:   Manage security-critical data to avoid storage space being exceeded  
Manage security-critical (TSF) data to ensure that the size of the data does not exceed the space allocated for storage of the data.
4. O.React\_Discovered\_Atk:   React to discovered attacks  
Implement automated notification or other reactions to the TSF-discovered attacks in an effort to identify attacks and to create an attack deterrent.
5. O.Data\_Imp\_Exp\_Control:   Data import/export to/from system control  
Protect data from being sent to erroneous places and more places external to the system than allowed by the organization's security policy. Conversely the import of data into the system should be protected from illicit information or information not allowed by the organization's security policy.
6. O.AuditLog:   Audit Log  
The TPS shall maintain the audit log

T.Hack\_Comm\_Eavesdrop:   Hacker eavesdrops on user data communications  
Hacker obtains user data by eavesdropping on communications lines.

In General, T.Hack\_Comm\_Eavesdrop is addressed by:

1. O.Data\_Exchange\_Conf:   Enforce data exchange confidentiality  
Protect user data confidentiality when exchanging data with a remote system.

T.Hack\_Crypto:   Cryptoanalysis for theft of information  
A hacker performs cryptoanalysis on encrypted data in order to recover message content.

In General, T.Hack\_Crypto is addressed by:

1. O.Crypto\_Data\_Sep:   Separation of cryptographic data  
Provide complete separation between plaintext and encrypted data and between data and keys. This requires separate channels and separate storage areas. The only place any data can pass between the plaintext and encrypted data modules is in the cryptographic engine. There should be no way for plaintext keys to reach either data module and no way for data to enter the key handling module.  
Encrypted keys can be handled as encrypted data, but with limited user access.
2. O.EMSEC\_Design:   Provide physical emanations security  
Design and build the system in such a way as to control the production of intelligible emanations within specified limits.
3. O.IntelEman\_Control:   Emanations control  
Limit system-produced intelligible emanations to within a specified limit.
4. O.IntelEman\_Contain:   Emanations containment  
Confine system-produced intelligible emanations to within a specified limit.
5. O.SpecRef:   Specification reference  
The TOE must provide all of the features and functions as specified in the T CPA specification.

6. O.Protected\_Capability: Protected Capability and shielded location  
The TOE must identify and protect capabilities as defined in the T CPA specification.

T.Hack\_Msg\_Data: Message content modification

A hacker modifies information intercepted from a communication link between two unsuspecting entities before passing it on, thereby deceiving the intended recipient.

In General, T.Hack\_Msg\_Data is addressed by:

1. O.Rcv\_MsgMod\_ID: Identify message modification in messages received  
The TSF recognizes changes to messages that occurred in transit, including insertion of spurious messages and deletion or replay of legitimate messages.
2. O.Snt\_MsgMod\_ID: Identify message modification in messages sent  
The TSF supports recognition of changes to transmitted messages that occurred in transit, including insertion of spurious messages and deletion or replay of legitimate messages.
3. O.TSF\_Rcv\_Err\_ID\_Loc: Local detection of received security-critical data modified in transit  
Identification by the system (TOE) of modification of security-critical (TSF) data occurring in transit from a remote trusted site must occur.
4. O.TSF\_Rcv\_Err\_ID\_Rem: Remote detection of received security-critical data modified in transit  
Identification by the remote site of the modification of security-critical (TSF) data occurring in transit from the remote site must occur.
5. O.TSF\_Snd\_Err\_ID\_Loc: Local detection of sent security-critical data modified in transit  
Identification of modification of security-critical (TSF) data occurring in transit to a remote site by the TSF must occur.
6. O.TSF\_Snd\_Err\_ID\_Rem: Remote detection of sent security-critical data modified in transit.  
Identification of modification of security-critical (TSF) data occurring in transit to a remote site by the remote site must occur.
7. O.AuditLog: Audit Log  
The TPS shall maintain the audit log

T.Hack\_Phys: Exploitation of vulnerabilities in the physical environment of the system

A hacker physically interacts with the system to exploit vulnerabilities in the physical environment, resulting in arbitrary security compromises.

In General, T.Hack\_Phys is addressed by:

1. O.EMSEC\_Design: Provide physical emanations security  
Design and build the system in such a way as to control the production of intelligible emanations within specified limits.

2. O.Tamper\_ID: Tamper detection  
Provide system features that detect physical tampering of a system component, and use those features to limit security breaches.
3. O.Tamper\_Resistance: Tamper resistance  
Prevent or resist physical tampering with specified system devices and components.
4. O.IntelEman\_Contain: Emanations containment  
Confine system-produced intelligible emanations to within a specified limit.
5. O.IntelEman\_Control: Emanations control  
Limit system-produced intelligible emanations to within a specified limit.

T.Hack\_Social\_Engineer: Social engineering

A hacker uses social engineering techniques to gain information about system entry, system use, system design, or system operation.

In General, T.Hack\_Social\_Engineer is addressed by:

1. O.Admin\_Guidance: Administrator guidance documentation  
Deter administrator errors by providing adequate administrator guidance.
2. O.User\_Guidance: User guidance documentation  
Provide documentation for the general user.

T.IdenClone: Identity cloning

Identities are unique keys that must remain protected by the TPM. Creating a copy of the identity breaks the uniqueness promise.

In General, T.IdenClone is addressed by:

1. O.TCPAIdentities: TCPA Identities  
The TOE must provide the ability to create, manage and use identities.
2. O.RootReporting: Reporting root of trust  
The reporting root of trust is the endorsement key. This provides a singular point that all challengers can rely on.
3. O.Crypto\_Data\_Sep: Separation of cryptographic data  
Provide complete separation between plaintext and encrypted data and between data and keys. This requires separate channels and separate storage areas. The only place any data can pass between the plaintext and encrypted data modules is in the cryptographic engine. There should be no way for plaintext keys to reach either data module and no way for data to enter the key handling module. Encrypted keys can be handled as encrypted data, but with limited user access.
4. O.Export\_Control: Sanitize data objects containing hidden or unused data  
Sanitize data objects that may contain hidden data when they are exported from the TOE in order to inhibit steganographic smuggling.
5. O.External\_Labels: Label or mark information for external systems  
Label or mark information for external systems to prevent the exchange of inappropriate data between systems.

6. O.Integ\_User\_Data\_Int: Protect user data during internal transfer  
Ensure the integrity of user data transferred internally within the system.
7. O.MetricReporting: Integrity metric reporting  
The TOE must report the values in the current PCR registers. The report may be digitally signed.

**T.IdenPKI: Identity PKI**

The identity creation process requires a PKI to certify the identity. This PKI must ensure the uniqueness of the identity and validate the endorsement key. Failure to properly perform these operations results in a bad identity.

In General, T.IdenPKI is addressed by:

1. O.TCPAIdentities: TCPA Identities  
The TOE must provide the ability to create, manage and use identities.
2. O.RootReporting: Reporting root of trust  
The reporting root of trust is the endorsement key. This provides a singular point that all challengers can rely on.
3. O.MetricReporting: Integrity metric reporting  
The TOE must report the values in the current PCR registers. The report may be digitally signed.

**T.Malicious\_Code: Malicious code exploitation**

An authorized user, IT system, or hacker downloads and executes malicious code, which causes abnormal processes that violate the integrity, availability, or confidentiality of system assets.

Coverage Rationale: An authorized user, IT system, or hacker downloads an object either deliberately or accidentally. The user does this primarily in order to gain assets that will assist in their job performance. The IT system may do this to meet informational requirements. The hacker may do this in an effort to satisfy destructive goals. The malicious code is then executed via a trigger mechanism. The trigger mechanism can be executed automatically after download, manually by the hacker, or unknowingly by the authorized user. The results of the attack affect the target system or any other system that the target system can influence.

In General, T.Malicious\_Code is addressed by:

1. O.Trusted\_Path: Provide a trusted path  
Provide a trusted path between the user and the system. Execution of a user-requested action must be made via a trusted path with the following properties:
  - \* The path is logically distinct from, and cannot be confused with other communication paths (by either the user or the system).
  - \* The path provides assured identification of its end points.
2. O.Admin\_Code\_Val: Administrative validation of executables  
Validate executable objects prior to allowing execution. Validation needs to be

done by someone with an expertise to recognize malicious code and the authority and means to prevent its execution.

3. O.Clean\_Obj\_Recovery: Object and data recovery free from malicious code  
Recover to a viable state after malicious code is introduced and damage occurs, removing the malicious code as part of the process.
4. O.Code\_Signing: Code signing and verification  
Check verification of signed downloaded code prior to execution. A well-known example is checking digital signatures on signed Java applets.
5. O.General\_Integ\_Checks: Periodically check integrity  
Provide periodic integrity checks on both system and user data.
6. O.Obj\_Protection: Object domain protection  
Require domain protection for objects. Specify object classes (domains), user groups, and operation classes. Use these to specify which operations may be performed on which objects by which users. Basically this controls what users can do in a given group.
7. O.Input\_Inspection: Require inspection for absence of malicious code.  
Require inspection of downloads/transfers.
8. O.AuditLog: Audit Log  
The TPS shall maintain the audit log

T.MeasureFalse: False integrity measurement

The entity or process providing the integrity measurement provides a false value.

In General, T.MeasureFalse is addressed by:

1. O.RootMeasurement: Measurement root of trust  
The entity that provides the base for measuring integrity values is the measurement root of trust. This entity on a PC would be the boot block or something similar.

T.OwnerMasquerade: Owner masquerade

At attacker can masquerade as the owner of either the TPM or an entity if they obtain the owner authorization data.

In General, T.OwnerMasquerade is addressed by:

1. O.SpecRef: Specification reference  
The TOE must provide all of the features and functions as specified in the TCPA specification.
2. O.TCPAIdentities: TCPA Identities  
The TOE must provide the ability to create, manage and use identities.
3. O.TCPAProtectedStorage: TCPA Protective Storage  
The TOE must provide a protected storage mechanism as defined in the specification section 3.6 and chapter 6.
4. O.Crypto\_Data\_Sep: Separation of cryptographic data  
Provide complete separation between plaintext and encrypted data and between

- data and keys. This requires separate channels and separate storage areas. The only place any data can pass between the plaintext and encrypted data modules is in the cryptographic engine. There should be no way for plaintext keys to reach either data module and no way for data to enter the key handling module. Encrypted keys can be handled as encrypted data, but with limited user access.
5. O.Crypto\_Dsgn\_Impl: Cryptographic Design and Implementation  
Minimize or even eliminate design and implementation errors in the cryptographic modules and functions.
  6. O.User\_Auth\_Management: User authorization management  
Manage and update user authorization and privilege data in accordance with organizational security and personnel policies.
  7. O.User\_Conf\_Prevention: Basic confidentiality-breach prevention  
Prevent unauthorized export of confidential information from the TOE with moderate effectiveness.

T.Power\_Disrupt: Unexpected disruption of system or component power  
A human or environmental agent disrupts power causing the system to lose information or security protection.

In General, T.Power\_Disrupt is addressed by:

1. O.Atomic\_Functions: Complete security functions or recover to previous state  
Recover automatically to a consistent, secure state if a security function does not complete successfully in the presence of certain types of failures.
2. O.Trusted\_Recovery: Trusted recovery of security functionality  
Recovery to a secure state, without security compromise, after a discontinuity of operations.

T.ProtStorAttribute: Protected storage attribute  
Each protected storage object has attributes that indicate its migration status, object type and source. Modification of these attributes allows the attacker to use the object in an unauthorized manner.

In General, T.ProtStorAttribute is addressed by:

1. O.Protected\_Capability: Protected Capability and shielded location  
The TOE must identify and protect capabilities as defined in the T CPA specification.
2. O.TCPAProtectedStorage: TCPA Protective Storage  
The TOE must provide a protected storage mechanism as defined in the specification section 3.6 and chapter 6.
3. O.Crypto\_Data\_Sep: Separation of cryptographic data  
Provide complete separation between plaintext and encrypted data and between data and keys. This requires separate channels and separate storage areas. The only place any data can pass between the plaintext and encrypted data modules is in the cryptographic engine. There should be no way for plaintext keys to reach



either data module and no way for data to enter the key handling module.

Encrypted keys can be handled as encrypted data, but with limited user access.

4. O.Data\_Exchange\_Conf: Enforce data exchange confidentiality  
Protect user data confidentiality when exchanging data with a remote system.
5. O.External\_Labels: Label or mark information for external systems  
Label or mark information for external systems to prevent the exchange of inappropriate data between systems.
6. O.Integ\_Data\_Mark\_Exp: Data marking integrity export  
Ensure that data markings are included with data that is exported to another trusted product.
7. O.Integ\_User\_Data\_Int: Protect user data during internal transfer  
Ensure the integrity of user data transferred internally within the system.
8. O.Integrity\_Data/SW: Integrity protection for user data and software  
Provide integrity protection for user data and software.
9. O.User\_Data\_Integrity: Integrity protection of stored user data  
Provide appropriate integrity protection for stored user data.

T.ProtStoreBackup: Protected storage backup

The protected storage backup mechanism must provide assurances that the migration and non-migration bits are properly followed. If they are not followed then non-migratable information may move from one system to another.

Coverage Rationale: The backup process can allow the transfer of information from one TPM to another if the migration status is not properly followed.

In General, T.ProtStoreBackup is addressed by:

1. O.TCPAProtectedStorage: TCPA Protective Storage  
The TOE must provide a protected storage mechanism as defined in the specification section 3.6 and chapter 6.
2. O.TCPAIdentities: TCPA Identities  
The TOE must provide the ability to create, manage and use identities.
3. O.Crypto\_Dsgn\_Impl: Cryptographic Design and Implementation  
Minimize or even eliminate design and implementation errors in the cryptographic modules and functions.
4. O.Export\_Control: Sanitize data objects containing hidden or unused data  
Sanitize data objects that may contain hidden data when they are exported from the TOE in order to inhibit steganographic smuggling.
5. O.External\_Labels: Label or mark information for external systems  
Label or mark information for external systems to prevent the exchange of inappropriate data between systems.
6. O.Integ\_Data\_Mark\_Exp: Data marking integrity export  
Ensure that data markings are included with data that is exported to another trusted product.
7. O.Trusted\_Recovery\_Doc: Documentation of untrusted data recovery  
Provide trusted recovery to ensure that data cannot be lost or misplaced. Any

circumstances which can cause untrusted recovery to be documented with mitigating procedures established.

8. O.Trusted\_Recovery: Trusted recovery of security functionality  
Recovery to a secure state, without security compromise, after a discontinuity of operations.

T.ProtStoreCrypto: Protected Storage Cryptography

Protected storage requires cryptography to protect the contents when the data is not inside the TPM. Failure of the cryptography exposes the data.

Coverage Rationale: The protected store must provide reasonable protections of the data using cryptography.

In General, T.ProtStoreCrypto is addressed by:

1. O.Protected\_Capability: Protected Capability and shielded location  
The TOE must identify and protect capabilities as defined in the TPCA specification.
2. O.TCPAProtectedStorage: TCPA Protective Storage  
The TOE must provide a protected storage mechanism as defined in the specification section 3.6 and chapter 6.
3. O.Crypto\_Data\_Sep: Separation of cryptographic data  
Provide complete separation between plaintext and encrypted data and between data and keys. This requires separate channels and separate storage areas. The only place any data can pass between the plaintext and encrypted data modules is in the cryptographic engine. There should be no way for plaintext keys to reach either data module and no way for data to enter the key handling module.  
Encrypted keys can be handled as encrypted data, but with limited user access.
4. O.Crypto\_Dsgn\_Impl: Cryptographic Design and Implementation  
Minimize or even eliminate design and implementation errors in the cryptographic modules and functions.
5. O.User\_Data\_Integrity: Integrity protection of stored user data  
Provide appropriate integrity protection for stored user data.

T.ProtStoreMaintenance: Protected storage maintenance

The protected storage maintenance feature allows for the cloning of a TPM. If this mechanism is abused then the attacker can make copies of TPM devices.

Coverage Rationale: When implemented the maintenance feature must be protected to eliminate the cloning problem.

In General, T.ProtStoreMaintenance is addressed by:

1. O.TCPAProtectedStorage: TCPA Protective Storage  
The TOE must provide a protected storage mechanism as defined in the specification section 3.6 and chapter 6.

2. O.Protected\_Capability: Protected Capability and shielded location  
The TOE must identify and protect capabilities as defined in the TCPA specification.
3. O.TCPAIdentities: TCPA Identities  
The TOE must provide the ability to create, manage and use identities.
4. O.Crypto\_Dsgn\_Impl: Cryptographic Design and Implementation  
Minimize or even eliminate design and implementation errors in the cryptographic modules and functions.
5. O.Data\_Exchange\_Conf: Enforce data exchange confidentiality  
Protect user data confidentiality when exchanging data with a remote system.
6. O.Export\_Control: Sanitize data objects containing hidden or unused data  
Sanitize data objects that may contain hidden data when they are exported from the TOE in order to inhibit steganographic smuggling.
7. O.External\_Labels: Label or mark information for external systems  
Label or mark information for external systems to prevent the exchange of inappropriate data between systems.
8. O.Integ\_Data\_Mark\_Exp: Data marking integrity export  
Ensure that data markings are included with data that is exported to another trusted product.
9. O.Trusted\_Recovery\_Doc: Documentation of untrusted data recovery  
Provide trusted recovery to ensure that data cannot be lost or misplaced. Any circumstances which can cause untrusted recovery to be documented with mitigating procedures established.
10. O.Trusted\_Recovery: Trusted recovery of security functionality  
Recovery to a secure state, without security compromise, after a discontinuity of operations.

T.Repudiate\_Receive: Recipient denies receiving information

The recipient of a message denies receiving the message, to avoid accountability for receiving the message or to avoid obligations incurred as a result of receiving the message.

In General, T.Repudiate\_Receive is addressed by:

1. O.NonRepud\_Assess\_Recd: Non-repudiation support for received information by a nonlocal sender's TSF  
Support nonrepudiation for received information by supporting remote handling of nonrepudiation evidence if needed.
2. O.NonRepud\_Gen\_Recd: Non-repudiation support for received information by the recipient's TSF  
Prevent a receiving user from avoiding accountability for receiving a message by providing evidence that the user received the message.

T.Repudiate\_Send: Sender denies sending information

The sender of a message denies sending the message to avoid accountability for sending the message or to avoid obligations incurred as a result of sending the message.

In General, T.Repudiate\_Send is addressed by:

1. O.NonRepud\_Assess\_Sent: Non-repudiation support for sent information by the nonlocal receiving TSF.  
Support nonrepudiation for sent information by supporting remote handling of nonrepudiation evidence if needed.
2. O.NonRepud\_Gen\_Sent: Non-repudiation support for sent information by the sender's TSF.  
Prevent a user from avoiding accountability for sending a message to a recipient at a different site by providing evidence that the user sent the message.

T.Repudiate\_Transact: A participant denies performing a transaction  
A participant in a transaction denies participation in the transaction to avoid accountability for the transaction or for resulting obligations.

In General, T.Repudiate\_Transact is addressed by:

1. O.NonRepud\_Assess\_Recd: Non-repudiation support for received information by a nonlocal sender's TSF  
Support nonrepudiation for received information by supporting remote handling of nonrepudiation evidence if needed.
2. O.NonRepud\_Assess\_Sent: Non-repudiation support for sent information by the nonlocal receiving TSF.  
Support nonrepudiation for sent information by supporting remote handling of nonrepudiation evidence if needed.
3. O.NonRepud\_Gen\_Recd: Non-repudiation support for received information by the recipient's TSF  
Prevent a receiving user from avoiding accountability for receiving a message by providing evidence that the user received the message.
4. O.NonRepud\_Gen\_Sent: Non-repudiation support for sent information by the sender's TSF.  
Prevent a user from avoiding accountability for sending a message to a recipient at a different site by providing evidence that the user sent the message.

T.SpecRef: Failure to follow specification  
The developer creating the TOE does not follow the TCPA specification and makes mistakes in implementation. This creates holes in the TOE that expose user and internal information.

Coverage Rationale: This threat provides broad coverage of the developer not following the specification. Additional threats provide specific details however this threat encompasses all failures to follow the specification.

In General, T.SpecRef is addressed by:

1. O.SpecRef: Specification reference  
The TOE must provide all of the features and functions as specified in the TCPA specification.
2. O.Integrity\_Data/SW: Integrity protection for user data and software  
Provide integrity protection for user data and software.
3. O.AuditLog: Audit Log  
The TPS shall maintain the audit log
4. O.MetricReporting: Integrity metric reporting  
The TOE must report the values in the current PCR registers. The report may be digitally signed.
5. O.Fail\_Secure: Preservation of secure state for failures in critical components  
Preserve the secure state of the system in the event of a secure component failure.
6. O.Fault\_Tolerance: Provide fault tolerant operations for critical components  
Provide fault tolerant operations for critical components and continue to operate in the presence of specific failures in one or more system components.
7. O.General\_Integ\_Checks: Periodically check integrity  
Provide periodic integrity checks on both system and user data.
8. O.Integ\_Data\_Mark\_Exp: Data marking integrity export  
Ensure that data markings are included with data that is exported to another trusted product.
9. O.Integ\_Sys\_Data\_Ext: Integrity of system data transferred externally  
Ensure the integrity of system data exchanged externally with another trusted product by using a protocol for data transfer that will permit error detection and correction.

This includes detecting and possibly correcting errors in data received and encoding outgoing data to make it possible for the receiver to detect and possibly correct errors. The method for detecting and correcting errors is based on some method (protocol) that is agreed upon by participating parties.

10. O.Integ\_Sys\_Data\_Int: Integrity of system data transferred internally  
Ensure the integrity of system data transferred internally.
11. O.Integ\_User\_Data\_Int: Protect user data during internal transfer  
Ensure the integrity of user data transferred internally within the system.
12. O.Integrity\_Data\_Rep: Integrity of system data replication  
Ensure that when system data replication occurs across the system the data is consistent for each replication.
13. O.Lifecycle\_Security: Lifecycle security  
Provide tools, techniques, and security employed during the development phase. Detect and resolve flaws during the operational phase. Provide safe destruction techniques.
14. O.Integrity\_Practice: Operational integrity system function testing  
Provide system functional tests to periodically test the integrity of the hardware and code running system functions.
15. O.Limit\_Actions\_Auth: Restrict actions before authentication  
Restrict the actions a user may perform before the TOE verifies the identity of the user.

16. O.Maintain\_Sec\_Domain: Maintain security domain  
Maintain at least one security domain for system (TOE) execution to protect the TOE from interference and tampering.
17. O.Malicious\_Code: Procedures for preventing malicious code  
Incorporate malicious code prevention procedures and mechanisms.
18. O.Manage\_Res\_Sec\_Attr: Manage resource security attributes  
Provide management on resource security attributes.
19. O.Manage\_TSF\_Data: Manage security-critical data to avoid storage space being exceeded  
Manage security-critical (TSF) data to ensure that the size of the data does not exceed the space allocated for storage of the data.
20. O.No\_Residual\_Info: Eliminate residual information  
Ensure there is no "object reuse;" i.e., ensure that there is no residual information in some information containers or system resources upon their reallocation to different users.

T.Spoofing: Legitimate system services are spoofed  
An attacker tricks users into interacting with spurious system services.

In General, T.Spoofing is addressed by:

1. O.Trusted\_Path: Provide a trusted path  
Provide a trusted path between the user and the system. Execution of a user-requested action must be made via a trusted path with the following properties:
  - \* The path is logically distinct from, and cannot be confused with other communication paths (by either the user or the system).
  - \* The path provides assured identification of its end points.

T.User\_Abuse\_Conf: Hostile user acts cause confidentiality breaches  
A user collects sensitive or proprietary information and removes it from the system.

In General, T.User\_Abuse\_Conf is addressed by:

1. O.Admin\_Code\_Val: Administrative validation of executables  
Validate executable objects prior to allowing execution. Validation needs to be done by someone with an expertise to recognize malicious code and the authority and means to prevent its execution.
2. O.Admin\_Guidance: Administrator guidance documentation  
Deter administrator errors by providing adequate administrator guidance.
3. O.Data\_Export\_Control: Control user data exportation  
Impose information control policies that do not allow export of specified data and/or export to specified locations.
4. O.Export\_Control: Sanitize data objects containing hidden or unused data  
Sanitize data objects that may contain hidden data when they are exported from the TOE in order to inhibit steganographic smuggling.

5. O.Standard\_Output\_Pres: Standard presentation of output values  
Present each possible output value in a standard form.
6. O.AuditLog: Audit Log  
The TPS shall maintain the audit log
7. O.SpecRef: Specification reference  
The TOE must provide all of the features and functions as specified in the TCPA specification.

T.User\_Collect: User abuses authorization to collect data  
User abuses granted authorizations to improperly collect sensitive or security-critical data.

In General, T.User\_Collect is addressed by:

1. O.Audit\_Generation: Audit records with identity  
Record in audit records: date and time of action, location of the action, and the entity responsible for the action.
2. O.Trusted\_Path: Provide a trusted path  
Provide a trusted path between the user and the system. Execution of a user-requested action must be made via a trusted path with the following properties:  
\* The path is logically distinct from, and cannot be confused with other communication paths (by either the user or the system).  
\* The path provides assured identification of its end points.
3. O.User\_Defined\_AC: User-defined access control  
Enforce an access control policy whereby users may determine who may access information they control.
4. O.Data\_Exchange\_Conf: Enforce data exchange confidentiality  
Protect user data confidentiality when exchanging data with a remote system.
5. O.Info\_Flow\_Control: System enforced information flow  
Enforce an information flow policy whereby users are constrained from allowing access to information they control, regardless of their intent (e.g., mandatory access control).  
This lattice property of security attributes is commonly associated with the U.S. DoD implementations of Mandatory Access Control (MAC).
6. O.Integ\_User\_Data\_Int: Protect user data during internal transfer  
Ensure the integrity of user data transferred internally within the system.
7. O.No\_Residual\_Info: Eliminate residual information  
Ensure there is no "object reuse;" i.e., ensure that there is no residual information in some information containers or system resources upon their reallocation to different users.
8. O.Security\_Roles: Security roles  
Maintain security-relevant roles and the association of users with those roles.

T.User\_Err\_Conf: User errors cause confidentiality breaches  
A user commits errors that cause information to be delivered to the wrong place or wrong person.

In General, T.User\_Err\_Conf is addressed by:

1. O.AC\_Label\_Export: Object security attributes and exportation  
Provide object security attributes in exported data with moderate to high effectiveness. The attributes are those associated with specific security function policies.
2. O.Crypto\_AC: Cryptographic access control policy  
Restrict user access to cryptographic IT assets in accordance with a specified user access control policy.
3. O.Crypto\_Import\_Export: Cryptographic import, export, and inter-TSF transfer  
Protect cryptographic data assets when they are being transmitted to and from the TOE, either through intervening untrusted components or directly to/from human users.
4. O.Crypto\_Key\_Man: Cryptographic Key Management  
Fully define cryptographic components, functions, and interfaces. Ensure appropriate protection for cryptographic keys throughout their lifecycle, covering generation, distribution, storage, use, and destruction.
5. O.User\_Conf\_Prevention: Basic confidentiality-breach prevention  
Prevent unauthorized export of confidential information from the TOE with moderate effectiveness.
6. O.SpecRef: Specification reference  
The TOE must provide all of the features and functions as specified in the T CPA specification.
7. O.TCPAIdentities: T CPA Identities  
The TOE must provide the ability to create, manage and use identities.

T.User\_Err\_Inaccess: User error makes data inaccessible  
A user accidentally deletes user data or changes system data rendering user data inaccessible.

In General, T.User\_Err\_Inaccess is addressed by:

1. O.User\_Guidance: User guidance documentation  
Provide documentation for the general user.
2. O.Security\_Attr\_Mgt: Manage security attributes  
Manage the initialization of, values for, and allowable operations on security attributes.

T.User\_Err\_Integrity: User errors cause integrity breaches  
A user commits errors that induce erroneous actions by the system and/or erroneous statements its users.

In General, T.User\_Err\_Integrity is addressed by:



1. O.AC\_Label\_Export: Object security attributes and exportation  
Provide object security attributes in exported data with moderate to high effectiveness. The attributes are those associated with specific security function policies.
2. O.Audit\_Generation: Audit records with identity  
Record in audit records: date and time of action, location of the action, and the entity responsible for the action.
3. O.Crypto\_Import\_Export: Cryptographic import, export, and inter-TSF transfer  
Protect cryptographic data assets when they are being transmitted to and from the TOE, either through intervening untrusted components or directly to/from human users.
4. O.Info\_Flow\_Control: System enforced information flow  
Enforce an information flow policy whereby users are constrained from allowing access to information they control, regardless of their intent (e.g., mandatory access control).  
This lattice property of security attributes is commonly associated with the U.S. DoD implementations of Mandatory Access Control (MAC).
5. O.User\_Defined\_AC: User-defined access control  
Enforce an access control policy whereby users may determine who may access information they control.
6. O.SpecRef: Specification reference  
The TOE must provide all of the features and functions as specified in the T CPA specification.

T.User\_Err\_Slf\_Protect: User errors undermine the system's security features  
A user commits errors that cause the system or one of its applications to undermine the system's security features.

In General, T.User\_Err\_Slf\_Protect is addressed by:

1. O.AC\_Label\_Export: Object security attributes and exportation  
Provide object security attributes in exported data with moderate to high effectiveness. The attributes are those associated with specific security function policies.
2. O.Obj\_Attr\_Integrity: Basic object attribute integrity  
Maintain object security attributes with moderate to high accuracy (under the guidance of qualified users).

T.User\_Misuse\_Avl\_Resc: User's misuse causes denial of service  
A user's unauthorized use of resources causes an undue burden on an affected resource.

In General, T.User\_Misuse\_Avl\_Resc is addressed by:

1. O.Audit\_Generation:    Audit records with identity  
Record in audit records: date and time of action, location of the action, and the entity responsible for the action.
2. O.Manage\_TSF\_Data:    Manage security-critical data to avoid storage space being exceeded  
Manage security-critical (TSF) data to ensure that the size of the data does not exceed the space allocated for storage of the data.
3. O.Tamper\_ID:    Tamper detection  
Provide system features that detect physical tampering of a system component, and use those features to limit security breaches.
4. O.Data\_Imp\_Exp\_Control:    Data import/export to/from system control  
Protect data from being sent to erroneous places and more places external to the system than allowed by the organization's security policy. Conversely the import of data into the system should be protected from illicit information or information not allowed by the organization's security policy.
5. O.Limit\_Comm\_Sessions:    Limit the number of user initiated communication sessions  
Provide mechanisms to limit the number of sessions that the user can initiate, if the user initiates multiple sessions that exceed the processors ability to perform in a reliable and efficient manner. These sessions could either be communication (TCP/IP) sessions or user login sessions.
6. O.Manage\_Res\_Sec\_Attr:    Manage resource security attributes  
Provide management on resource security attributes.
7. O.Tamper\_Resistance:    Tamper resistance  
Prevent or resist physical tampering with specified system devices and components.
8. O.SpecRef:    Specification reference  
The TOE must provide all of the features and functions as specified in the T CPA specification.

T.User\_Modify:    User abuses authorization to modify data  
A user abuses granted authorizations to improperly change or destroy sensitive or security-critical data.

In General, T.User\_Modify is addressed by:

1. O.Audit\_Generation:    Audit records with identity  
Record in audit records: date and time of action, location of the action, and the entity responsible for the action.
2. O.Config\_Management:    Implement operational configuration management  
Implement a configuration management plan. Implement configuration management to assure storage integrity, identification of system connectivity (software, hardware, and firmware), and identification of system components (software, hardware, and firmware).
3. O.General\_Integ\_Checks:    Periodically check integrity  
Provide periodic integrity checks on both system and user data.

4. O.Info\_Flow\_Control: System enforced information flow  
Enforce an information flow policy whereby users are constrained from allowing access to information they control, regardless of their intent (e.g., mandatory access control).  
This lattice property of security attributes is commonly associated with the U.S. DoD implementations of Mandatory Access Control (MAC).
5. O.Integrity\_Practice: Operational integrity system function testing  
Provide system functional tests to periodically test the integrity of the hardware and code running system functions.
6. O.Security\_Data\_Mgt: Manage security-critical data  
Manage the initialization of, limits on, and allowable operations on security-critical data.
7. O.Security\_Roles: Security roles  
Maintain security-relevant roles and the association of users with those roles.
8. O.User\_Defined\_AC: User-defined access control  
Enforce an access control policy whereby users may determine who may access information they control.
9. O.Audit\_Protect: Protect stored audit records  
Protect audit records against unauthorized access, modification, or deletion to ensure accountability of user actions.
10. O.Integ\_Sys\_Data\_Int: Integrity of system data transferred internally  
Ensure the integrity of system data transferred internally.
11. O.Maintain\_Sec\_Domain: Maintain security domain  
Maintain at least one security domain for system (TOE) execution to protect the TOE from interference and tampering.
12. O.SpecRef: Specification reference  
The TOE must provide all of the features and functions as specified in the T CPA specification.

T.User\_Send: User abuses authorization to send data  
A user abuses granted authorizations to improperly send sensitive or security-critical data.

In General, T.User\_Send is addressed by:

1. O.Admin\_Code\_Val: Administrative validation of executables  
Validate executable objects prior to allowing execution. Validation needs to be done by someone with an expertise to recognize malicious code and the authority and means to prevent its execution.
2. O.Audit\_Generation: Audit records with identity  
Record in audit records: date and time of action, location of the action, and the entity responsible for the action.
3. O.Export\_Control: Sanitize data objects containing hidden or unused data  
Sanitize data objects that may contain hidden data when they are exported from the TOE in order to inhibit steganographic smuggling.

4. O.Standard\_Output\_Pres: Standard presentation of output values  
Present each possible output value in a standard form.
5. O.Integ\_Data\_Mark\_Exp: Data marking integrity export  
Ensure that data markings are included with data that is exported to another trusted product.
6. O.SpecRef: Specification reference  
The TOE must provide all of the features and functions as specified in the TCPA specification.

## 6.3 - Security Requirements Rationale

Demonstrate that the set of security requirements identified in Section 5 is suitable to meet the security objectives identified in Section 4.

### 6.3.1 - Functional Security Requirements Rationale

**Table 6-3 Functional Component to Security Objective Mapping**

Objectives	Requirements
O.AC_Label_Export	FDP_ACF.1, FDP_ETC.2, FDP_ACC.2
O.Admin_Code_Val	FDP_ACF.1, FMT_MSA.1, FPT_TST.1, FDP_ACC.2, FDP_SDI.2
O.Admin_Guidance	AGD_ADM.1
O.Apply_Code_Fixes	ADO_DEL.1, AGD_ADM.1, FMT_MOF.1, FMT_MSA.1
O.Atomic_Functions	FPT_RCV.4
O.AuditLog	FAU_GEN.1, FAU_SEL.1, FAU_SAR.1, FAU_SAA.1
O.Audit_Generation	FAU_GEN.1
O.Audit_Protect	FAU_STG.1
O.Change_Control_Users	FDP_DAU.1
O.Clean_Obj_Recovery	FDP_ETC.2, FDP_ITC.1, FDP_ROL.1, FMT_MOF.1, FPT_TST.1
O.Code_Signing	FDP_ETC.2, FDP_ITC.2, FDP_UIT.1
O.Config_Management	FMT_MOF.1, FMT_MTD.1

O.Crypto_AC	FDP_ACF.1, FDP_ACC.2
O.Crypto_Data_Sep	FPT_AMT.1, FPT_SEP.2
O.Crypto_Dsgn_Impl	ADV_HLD.2, ADV_RCR.1
O.Crypto_Import_Export	AGD_ADM.1, AGD_USR.1, FDP_ETC.2, FDP_ITC.1, FDP_ITC.2, FTP_ITC.1
O.Crypto_Key_Man	ADV_FSP.1, ADV_SPM.1, AVA_VLA.1, FCS_CKM.1, FCS_CKM.4, FDP_ACF.1, FDP_ITC.1, FMT_MSA.1, FMT_MTD.1, FDP_ACC.2, FPT_SEP.2
O.Crypto_Modular_Dsgn	ADV_FSP.1
O.Crypto_Operation	ADV_FSP.1, ADV_SPM.1, FCS_COP.1
O.Crypto_Self_Test	FPT_AMT.1, FPT_FLS.1, FPT_TST.1, FDP_SDI.2
O.Crypto_Test_Reqs	FPT_AMT.1, ATE_DPT.1, ATE_FUN.1, AVA_VLA.1
O.Data_Exchange_Conf	FCS_COP.1
O.Data_Export_Control	FDP_ETC.2
O.Data_Imp_Exp_Control	FDP_ETC.2, FDP_IFF.1, FDP_ITC.2
O.EMSEC_Design	ADV_FSP.1, ADV_HLD.2, ATE_COV.2, ATE_DPT.1, ATE_FUN.1, ATE_IND.2, FPT_PHP_EMSEC_Design
O.Export_Control	FDP_ACC.2
O.External_Labels	FDP_ETC.2, FDP_ITC.2
O.Fail_Secure	FPT_FLS.1
O.Fault_Tolerance	FRU_FLT.1
O.General_Integ_Checks	FPT_TST.1, FDP_SDI.2
O.Hack_Limit_Sessions	AGD_ADM.1, FMT_MSA.1
O.Info_Flow_Control	FDP_IFC.2, FDP_IFF.1
O.Input_Inspection	FDP_ACF.1, FDP_ITC.1, FDP_ACC.2
O.Integ_Data_Mark_Exp	FDP_ETC.2

O.Integ_Sys_Data_Ext	FPT_ITI.1
O.Integ_Sys_Data_Int	FPT_SSP.2, FPT_ITT.2
O.Integ_User_Data_Int	FDP_ITT.2
O.Integrity_Data/SW	FDP_SDI.2
O.Integrity_Data_Rep	FPT_TRC.1
O.Integrity_Practice	FPT_AMT.1, FPT_TST.1
O.IntelEman_Contain	ATE_IND.2, ADV_HLD.2, ADV_FSP.1
O.IntelEman_Control	ATE_IND.2, ADV_FSP.1, ADV_HLD.2
O.Lifecycle_Security	ALC_DVS.1, ALC_LCD.1
O.Limit_Actions_Auth	FIA_UAU.2
O.Limit_Comm_Sessions	FTA_MCS.1
O.Maintain_Sec_Domain	FPT_SEP.2
O.Malicious_Code	FDP_ITC.1, FPT_AMT.1, FPT_PHP.1, FPT_TST.1
O.Manage_Res_Sec_Attr	AGD_USR.1, FAU_GEN.1, FMT_MSA.1
O.Manage_TSF_Data	FMT_MTD.2
O.MessageAuthentication	FCO_NRO.1, FCO_NRO.2, FCO_NRR.1, FCO_NRR.2, FCS_COP.1, FIA_AFL.1, FIA_UAU.2, FIA_UAU.4, FIA_UID.2, FDP_ACC.2, FDP_ACF.1, FDP_DAU.1, FPT_RPL.1
O.MetricReporting	ADV_HLD.2, ADV_FSP.1
O.NoBore	ADV_FSP.1, ADV_HLD.2, ADV_SPM.1
O.No_Residual_Info	FDP_RIP.1, FDP_RIP.2
O.NonRepud_Assess_Recd	AGD_ADM.1, AGD_USR.1, FCO_NRR.1, FMT_MOF.1
O.NonRepud_Assess_Sent	AGD_ADM.1, AGD_USR.1, FCO_NRO.1, FMT_MOF.1
O.NonRepud_Gen_Recd	AGD_ADM.1, FCO_NRR.1, FCO_NRR.2, FMT_MOF.1

O.NonRepud_Gen_Sent	AGD_ADM.1, FCO_NRO.1, FCO_NRO.2, FMT_MOF.1
O.Obj_Attr_Integrity	FDP_ACF.1, FMT_MSA.1, FMT_MSA.2, FMT_MSA.3, FDP_ACC.2, FMT_SMR.2
O.Obj_Protection	FDP_ACF.1, FMT_MSA.3, FDP_ACC.2
O.Prevent_Link	FPR_UNL.1
O.Protected_Capability	ADV_HLD.2, ADV_FSP.1, FCS_COP.1, FCS_CKM.1, FCS_CKM.4, FDP_ITT.4
O.Rcv_MsgMod_ID	FDP_UIT.1
O.React_Discovered_Atk	AGD_ADM.1, FAU_ARP.1
O.RootMeasurement	ADV_FSP.1, ADV_HLD.2
O.RootReporting	ADV_FSP.1, ADV_HLD.2, ADV_SPM.1
O.SecureManufacturing	ALC_DVS.1
O.Secure_State	FPT_FLS.1, FPT_RCV.3, FPT_RCV.4
O.Security_Attr_Mgt	FMT_MSA.1, FMT_MSA.2, FMT_MSA.3
O.Security_Data_Mgt	FMT_MTD.1, FMT_MTD.2, FMT_MTD.3
O.Security_Func_Mgt	FMT_MOF.1
O.Security_Roles	FMT_SMR.2
O.Snt_MsgMod_ID	FDP_UIT.1
O.Source_Code_Exam	ADV_RCR.1
O.SpecRef	FCS_CKM.1, FCS_CKM.4, FPT_ITT.2, FPT_PHP.1, FPT_PHP.3, FPT_RPL.1, ACM_SCP.1, ADO_DEL.1, ADO_IGS.1, ADV_FSP.1, ADV_HLD.2, ADV_RCR.1, ADV_SPM.1, AGD_ADM.1, AGD_USR.1, ALC_DVS.1, ALC_LCD.1, AVA_MSU.1, AVA_SOF.1, AVA_VLA.1, ATE_DPT.1, ATE_FUN.1, ATE_IND.2, ACM_CAP.3, ATE_COV.2, FDP_ITT.4, FPT_TDC.1, FRU_PRS.1
O.Standard_Output_Pres	ADV_FSP.1, ADV_HLD.2
O.Storage_Integrity	FDP_SDI.2

O.Sys_Assur_HW/SW/FW	ATE_FUN.1, FPT_TST.1
O.Sys_Backup_Procs	FPT_RCV.3
O.Sys_Backup_Verify	FPT_AMT.1, FPT_PHP.1, FPT_TST.1
O.Sys_Self_Protection	FPT_SEP.2
O.TCPAIdentities	FCS_CKM.1, FCS_CKM.4
O.TCPAProtectedStorage	FDP_ACC.2, FDP_ACF.1, FDP_DAU.1, FDP_ETC.2, FDP_IFC.2, FDP_IFF.1, FDP_ITC.1, FDP_ITC.2, FDP_ITT.2, FDP_RIP.1, FDP_RIP.2, FDP_ROL.1, FDP_SDI.2, FDP_UCT.1, FDP_UIT.1, FIA_AFL.1, FIA_UAU.2, FPR_ANO.2
O.TSF_Rcv_Err_ID_Loc	FPT_ITI.1
O.TSF_Rcv_Err_ID_Rem	FPT_ITI.1
O.TSF_Snd_Err_ID_Loc	FPT_ITI.1
O.TSF_Snd_Err_ID_Rem	FPT_ITI.1
O.Tamper_ID	AGD_ADM.1, AGD_USR.1, FPT_PHP.1
O.Tamper_Resistance	FPT_PHP.3
O.Trusted_Path	FTP_ITC.1, FTP_TRP.1
O.Trusted_Recovery	FPT_RCV.3
O.Trusted_Recovery_Doc	AGD_ADM.1
O.User_Auth_Management	AGD_ADM.1, AGD_USR.1, FMT_MSA.1, FMT_REV.1
O.User_Conf_Prevention	FDP_ACF.1, FDP_IFF.1, FDP_ACC.2
O.User_Data_Integrity	FDP_SDI.2
O.User_Data_Transfer	FDP_ITT.2
O.User_Defined_AC	FDP_ACC.2, FDP_ACF.1
O.User_Guidance	AGD_USR.1
Security Objectives	ATE_COV.2



**O.AC\_Label\_Export:** Object security attributes and exportation  
Provide object security attributes in exported data with moderate to high effectiveness.  
The attributes are those associated with specific security function policies.

O.AC\_Label\_Export is implemented in the TOE by:

1. FDP\_ACF.1: Security attribute based access control
2. FDP\_ETC.2: Export of user data with security attributes

Component Application: Apply using attributes relevant to labeling on export.

3. FDP\_ACC.2: Complete access control

**O.Admin\_Code\_Val:** Administrative validation of executables  
Validate executable objects prior to allowing execution. Validation needs to be done by someone with an expertise to recognize malicious code and the authority and means to prevent its execution.

Implementation Application: Choose FDP\_ACC.1, FDP\_AFF.1, FDP\_ACC.1, FMT\_MSA.1, and FPT\_TST.1 to implement the objective.

O.Admin\_Code\_Val is implemented in the TOE by:

1. FDP\_ACF.1: Security attribute based access control
2. FMT\_MSA.1: Management of security attributes
3. FPT\_TST.1: TSF testing
4. FDP\_ACC.2: Complete access control
5. FDP\_SDI.2: Stored data integrity monitoring and action

**O.Admin\_Guidance:** Administrator guidance documentation  
Deter administrator errors by providing adequate administrator guidance.

O.Admin\_Guidance is implemented in the TOE by:

1. AGD\_ADM.1: Administrator guidance

**O.Apply\_Code\_Fixes:** Apply patches to fix the code  
Apply patches to fix the code when vulnerabilities in code allow unauthorized and undiscovered access.

Implementation Application: Any of the ALC\_FLR components may be used when implementing this objective, with resultant differences in effectiveness of the objective.

Currently, the implementing components apply primarily to the TOE environment. However, the PP author may wish to also consider requirements for the TSF to actively support maintenance via safe installation programs and/or TOE version control mechanisms.

O.Apply\_Code\_Fixes is implemented in the TOE by:

1. ADO\_DEL.1: Delivery procedures
2. AGD\_ADM.1: Administrator guidance

Component Application: Administrators in the TSF-maintenance role have a responsibility to modify the security behavior of the system by applying developer-supplied code fixes according to developer-specified procedures. TSF-maintenance administrators shall make sure that security attributes are properly maintained and upgraded as code fixes are applied.

3. FMT\_MOF.1: Management of security functions behaviour

Component Application: Create a TSF-maintenance role with the ability to modify the behavior of arbitrary TSF functions.

4. FMT\_MSA.1: Management of security attributes

Component Application: Security policies enforced by the TSF need to allow necessary maintenance of security attributes by TSF-maintenance administrators.

O.Atomic\_Functions: Complete security functions or recover to previous state. Recover automatically to a consistent, secure state if a security function does not complete successfully in the presence of certain types of failures.

Implementation Application: Choose FPT\_RCV.4 to implement the objective.

O.Atomic\_Functions is implemented in the TOE by:

1. FPT\_RCV.4: Function recovery

Component Application: To support the implementation of this requirement, the developer must provide a definition of "secure state" so that the requirement can be evaluated. This definition can be provided through the associated dependency on ADV\_SPM.1.

Component Application Rationale: The definition of a "secure state" should

provided by the security model documentation (ADV\_SPM.1) when a model is required by the PP. If the requirement for a security model is not present, it is still necessary for the developer to provide this definition in some manner. How this definition is provided should be identified in the rationale that defends the exclusion of ADV\_SPM.1.

O.AuditLog:     Audit Log  
The TPS shall maintain the audit log

O.AuditLog is implemented in the TOE by:

1. FAU\_GEN.1:     Audit data generation
2. FAU\_SEL.1:     Selective audit

O.AuditLog is implemented in the IT environment by:

1. FAU\_SAR.1:     Audit review
2. FAU\_SAA.1:     Potential violation analysis

O.Audit\_Generation:     Audit records with identity  
Record in audit records: date and time of action, location of the action, and the entity responsible for the action.

O.Audit\_Generation is implemented in the TOE by:

1. FAU\_GEN.1:     Audit data generation

Component Application:     Level of auditing: This should generally not be determined on the basis of a single application of this component, but through consideration of audit requirements in their entirety.

O.Audit\_Protect:     Protect stored audit records  
Protect audit records against unauthorized access, modification, or deletion to ensure accountability of user actions.

Implementation Application:     There are two variants of this objective (choose one):  
[Basic]: Choose FAU\_STG.1.  
[Availability]: Choose FAU\_STG.2.

O.Audit\_Protect is implemented in the IT environment by:

1. FAU\_STG.1: Protected audit trail storage

Component Application: Select prevent.

O.Change\_Control\_Users: User notification of data content changes  
Notify users of changes to data content in order to make any adjustments to their own data.

O.Change\_Control\_Users is implemented in the TOE by:

1. FDP\_DAU.1: Basic data authentication

O.Clean\_Obj\_Recovery: Object and data recovery free from malicious code  
Recover to a viable state after malicious code is introduced and damage occurs, removing the malicious code as part of the process.

O.Clean\_Obj\_Recovery is implemented in the TOE by:

1. FDP\_ETC.2: Export of user data with security attributes

Component Application: Apply this in such a way as to make sure data needed for backup is reliably exported and marked to prevent confusing it with invalid backup data. In the case of an intelligent remote backup device, this component may also be applied to control information used to retrieve backup data, to prevent spoofing.

2. FDP\_ITC.1: Import of user data without security attributes

Component Application: Apply this in such a way as to make sure that the restored backup data is not corrupted.

3. FDP\_ROL.1: Basic rollback

Component Application: Apply this component to cover operations that may corrupt an object. For example, operations performed by malicious code that has been accidentally loaded onto a system.

4. FMT\_MOF.1: Management of security functions behaviour

Component Application: Restrict the use of rollback facilities to an appropriate administrative role.

5. FPT\_TST.1: TSF testing

Component Application: Specify restoration of TSF code and data as conditions

under which self test should occur, in order to check that the TSF has been restored properly.

**O.Code\_Signing:** Code signing and verification  
Check verification of signed downloaded code prior to execution. A well-known example is checking digital signatures on signed Java applets.

O.Code\_Signing is implemented in the TOE by:

1. FDP\_ETC.2: Export of user data with security attributes

Component Application: Environment: Apply FDP\_ETC.2 to require signatures on code exported by the developer.

2. FDP\_ITC.2: Import of user data with security attributes

Component Application: Apply this component to require checking of digital signatures on imported code.

3. FDP\_UIT.1: Data exchange integrity

Component Application: Allocation: FDP\_UIT.1.2 is for the TOE; FDP\_UIT.1.1 is for the Environment. Apply this element to specify that the digital signature process is effective; i.e. prevents spoofing or masquerading.

**O.Config\_Management:** Implement operational configuration management  
Implement a configuration management plan. Implement configuration management to assure storage integrity, identification of system connectivity (software, hardware, and firmware), and identification of system components (software, hardware, and firmware).

O.Config\_Management is implemented in the TOE by:

1. FMT\_MOF.1: Management of security functions behaviour

Component Application: Apply this component in such a way as to achieve operational configuration management. Use as a model for this, the corresponding configuration management requirements for the development environment, e.g. ACM\_AUT.1.

2. FMT\_MTD.1: Management of TSF data

Component Application: Apply this component in such a way as to achieve operational configuration management. Use as a model for this, the corresponding configuration management requirements for the development environment, e.g. ACM\_AUT.1.

**O.Crypto\_AC:** Cryptographic access control policy

Restrict user access to cryptographic IT assets in accordance with a specified user access control policy.

O.Crypto\_AC is implemented in the TOE by:

1. FDP\_ACF.1: Security attribute based access control

**Component Application:** Specify attributes for objects, such as the object's function (e.g., whether it is encrypted), associated roles, user ownership, and validity period (if appropriate). For the special case of cryptographic keys, specify additional attributes such as key type (e.g. public, private, secret), validity period, and intended use (e.g. digital signature, key encryption, key agreement, data encryption).

Specify rules governing access to cryptographic assets, including rules governing the allowed use of cryptographic operations on relevant objects (e.g., plaintext, ciphertext, red or black cryptographic keys).

2. FDP\_ACC.2: Complete access control

**O.Crypto\_Data\_Sep:** Separation of cryptographic data

Provide complete separation between plaintext and encrypted data and between data and keys. This requires separate channels and separate storage areas. The only place any data can pass between the plaintext and encrypted data modules is in the cryptographic engine. There should be no way for plaintext keys to reach either data module and no way for data to enter the key handling module. Encrypted keys can be handled as encrypted data, but with limited user access.

**Implementation Application:** Choose ADV\_INT.1, FPT\_SEP.2, and FDP\_IFF.3; apply using a technical policy on separation of encrypted and plaintext data (sometimes referred to as red-black separation).

O.Crypto\_Data\_Sep is implemented in the TOE by:

1. FPT\_AMT.1: Abstract machine testing
2. FPT\_SEP.2: SFP domain separation

**Component Application:** Specify policies concerning red/black data separation and data/key separation.

**O.Crypto\_Dsgn\_Impl:** Cryptographic Design and Implementation

Minimize or even eliminate design and implementation errors in the cryptographic modules and functions.

O.Crypto\_Dsgn\_Impl is implemented in the TOE by:

1. ADV\_HLD.2: Security enforcing high-level design

Component Application: Require description of the cryptographic portion of the TSF in terms of major structural units (i.e. sub-systems) and relating these units to the functions that they contain. Require distinguishing the cryptographic boundary of the TOE from the overall TOE boundary.

2. ADV\_RCR.1: Informal correspondence demonstration

Component Application: Require demonstration of the correspondence between various representations of the cryptographic design.

O.Crypto\_Import\_Export: Cryptographic import, export, and inter-TSF transfer Protect cryptographic data assets when they are being transmitted to and from the TOE, either through intervening untrusted components or directly to/from human users.

Implementation Application: Three different implementations may achieve the intent of this objective (and combinations are possible):

1. Use FTP\_TRP.1 and/or FTP\_TRC.1 to specify trusted paths and channels for communicating cryptographic assets.
2. Require a separate physical port for input and output of such information; use FDP\_ETC.1 and FDP\_ITC.1 for this.
3. Require security labeling of cryptographic data; in this case, use FDP\_ETC.2 and FDP\_ITC.2 instead.

All three implementations assume user cognizance. Specify this using AGD\_ADM.1 and/or AGD\_USR.1.

O.Crypto\_Import\_Export is implemented in the TOE by:

1. AGD\_ADM.1: Administrator guidance

Component Application: Instruct administrators on the sensitivity of cryptographic assets and on the importance of not mix them with other information.

2. AGD\_USR.1: User guidance

Component Application: Instruct users on the sensitivity of cryptographic assets and on the importance of not mix them with other information.

3. FDP\_ETC.2: Export of user data with security attributes

Component Application: Specify constraints on I/O devices to protect them from improper distribution. Specify labels to facilitate separation of cryptographic assets from other assets.

4. FDP\_ITC.1: Import of user data without security attributes

Component Application: Specify constraints to avoid loading of weakened or corrupted cryptographic assets.

5. FDP\_ITC.2: Import of user data with security attributes

Component Application: Specify constraints to avoid loading of weakened or corrupted cryptographic assets.

6. FTP\_ITC.1: Inter-TSF trusted channel

Component Application: Use to facilitate protected communication of cryptographic assets.

#### O.Crypto\_Key\_Man: Cryptographic Key Management

Fully define cryptographic components, functions, and interfaces. Ensure appropriate protection for cryptographic keys throughout their lifecycle, covering generation, distribution, storage, use, and destruction.

Implementation Application: Cryptographic function may be specified either directly via functional requirements or indirectly via assurance requirements. Directly specify key management using components of the FCS\_CKM family, with the following additions, depending on the nature of the keys:

protection of stored user keys: use FDP\_ACC and FDP\_ACF, refine AVA\_VLA.1

user key attributes: use FMT\_MSA, FDP\_ACC, and FDP\_ACF.

user key entry: use FDP\_ITC.

protection of stored TSF keys: use FPT\_SEP.

TSF key entry: use FMT\_MTD.

Alternatively or additionally, require the TOE developer to specify cryptographic key management by using ADV\_FSP and ADV\_SPM.

O.Crypto\_Key\_Man is implemented in the TOE by:

1. ADV\_FSP.1: Informal functional specification

Component Application: Require a high level description of the user-visible interface and behaviour of the cryptographic components of the TSF.



2. ADV\_SPM.1: Informal TOE security policy model

Component Application: Include if there is a requirement for security policy model.

3. AVA\_VLA.1: Developer vulnerability analysis

Component Application: Check for ways to evade cryptographic access control policy. Specifically, consider ways for unauthorized personnel to recover plaintext cryptographic keys and authentication data (e.g., from audit records, temporary objects, free storage areas).

4. FCS\_CKM.1: Cryptographic key generation

Component Application: Allocate to the TOE if the TOE performs key generation.

5. FCS\_CKM.4: Cryptographic key destruction

Component Application: Allocate to the TOE if the TOE performs any cryptographic operations.

6. FDP\_ACF.1: Security attribute based access control

Component Application: Using the access control policy for protection of cryptographic keys, require that the TSF store keys in encrypted form and protect them from unauthorized use.

7. FDP\_ITC.1: Import of user data without security attributes

Component Application: Use the access control policy for protection of cryptographic keys to control key entry. Specify, e.g., whether keys are to be entered in unencrypted, encrypted, or split-knowledge forms.

8. FMT\_MSA.1: Management of security attributes

Component Application: Specify management of cryptographic key attributes such as key type (e.g. public, private, secret), validity period, and intended use (e.g. digital signature, key encryption, key agreement, data encryption).

9. FMT\_MTD.1: Management of TSF data

Component Application: Cover import/entry of TSF cryptographic keys, key escrow (if relevant), key generation, etc.

10. FDP\_ACC.2: Complete access control

11. FPT\_SEP.2: SFP domain separation

O.Crypto\_Modular\_Dsgn: Cryptographic Modular Design

Prevent errors in one part of the TOE from influencing other parts, especially cryptographic parts. To this end, noncryptographic I/O paths must be well defined and logically independent of circuitry and processes performing key generation, manual key entry, key zeroising, and similar key-related operations.

Implementation Application: ---

This objective helps insure that the TOE is designed using sound engineering principles and, hence, that data is accessed only by the components of the TOE that need it. The low-level design requirements that support this objective can be used to ensure that the inputs, outputs, plaintexts, and ciphertexts are accessed only by components of the TOE that need them.

O.Crypto\_Modular\_Dsgn is implemented in the TOE by:

1. ADV\_FSP.1: Informal functional specification

Component Application: Require specification of the presentation of syntax and semantics of all external cryptographic TSF interfaces. Require evidence demonstrating that the cryptographic functions in the TSF are completely represented.

O.Crypto\_Operation: Cryptographic function definition

Cryptographic components, functions, and interfaces shall be fully defined.

Implementation Application: Cryptographic function may be specified either directly via functional requirements or indirectly via assurance requirements. Directly specify cryptographic operations using components of the FCS\_COP family. Alternatively or additionally, require the TOE developer to specify cryptographic operation by using ADV\_FSP and ADV\_SPM.

O.Crypto\_Operation is implemented in the TOE by:

1. ADV\_FSP.1: Informal functional specification

Component Application: Require a high level description of the user-visible interface and behaviour of the cryptographic components of the TSF.

2. ADV\_SPM.1: Informal TOE security policy model

Component Application: Include if there is a requirement for security policy model.

3. FCS\_COP.1: Cryptographic operation

Component Application: Include if the TOE performs any perform the cryptographic operations. Note that encryption of keys while in storage is one of the cryptographic operations that must be listed.

O.Crypto\_Self\_Test: Cryptographic self test  
Provide the ability to verify that the cryptographic functions operate as designed.

O.Crypto\_Self\_Test is implemented in the TOE by:

1. FPT\_AMT.1: Abstract machine testing

Component Application: Apply to testing the cryptographic portion of the underlying abstract state machine.

2. FPT\_FLS.1: Failure with preservation of secure state
3. FPT\_TST.1: TSF testing

Component Application: Refine to cover tests to ensure that the cryptographic functions are operating correctly. Tests conducted during start-up and/or periodically may include known-answer tests of cryptographic operations, as well as statistical tests on random number generators. Additional tests may involve generation of private / public key pairs, pair-wise consistency tests of encryption and decryption, key-entry tests, and key integrity tests.

4. FDP\_SDI.2: Stored data integrity monitoring and action

O.Crypto\_Test\_Reqs: Test cryptographic functionality  
Test cryptographic operation and key management.

Implementation Application: Testing requirements should be selected based on the sensitivity of the application and vulnerability of the TOE to attack. In particular, greater depth of testing may provide better assurance of correct function, and may well be feasible, as cryptographic functions are often fairly simple and modular.

O.Crypto\_Test\_Reqs is implemented in the TOE by:

1. FPT\_AMT.1: Abstract machine testing
2. ATE\_DPT.1: Testing: high-level design
3. ATE\_FUN.1: Functional testing
4. AVA\_VLA.1: Developer vulnerability analysis

Component Application: Provide independent testing of cryptographic functionality.

**O.Data\_Exchange\_Conf:** Enforce data exchange confidentiality  
Protect user data confidentiality when exchanging data with a remote system.

O.Data\_Exchange\_Conf is implemented in the TOE by:

1. **FCS\_COP.1:** Cryptographic operation

**Component Application:** Complete this component's operations in a manner compatible with the associated FCS\_CKM components.

**List of cryptographic operations:** a function for encrypting and/or decrypting user data that is exported and/or imported

**O.Data\_Export\_Control:** Control user data exportation  
Impose information control policies that do not allow export of specified data and/or export to specified locations.

**Implementation Application:** There are two variants of this objective (choose one):  
[Unmarked]: Choose FDP\_ETC.1.  
[Marked]: Choose FDP\_ETC.2.

The choice of whether to export data with or without security attributes will depend on several considerations. The predominant considerations are the potential and actual uses of the security attributes in the destination environment. The less potential for use in the destination environment, the less motivation to mark exported data with security attributes, even if such attributes are maintained internally.

---

The [Unmarked] variant applies a policy enforcement to the export of user data, but does not require security attributes to be associated with the exported data itself. The term "unmarked" indicates that the data is not associated with any security attributes whatsoever, avoiding possible misconceptions that might be implied by the term "unlabeled."

The [Marked] variant applies policy enforcement to the export of user data. In addition, the implementation provides security attributes that are associated, with moderate to high effectiveness, with instances of exported data. The attributes are those associated with specific security function policies. The term "marked" indicates that the security attributes may not actually represent security labels in its common usage.

O.Data\_Export\_Control is implemented in the TOE by:

1. **FDP\_ETC.2:** Export of user data with security attributes

**Component Application Rationale:** Explain how the enforcement of the

identified security policy for exported data achieves some security protection need that caused the inclusion of the related objective.

The binding of security attributes to exported data strongly implies some security-relevant use of those attributes in the destination environment. Note that in many cases the exported environment is local, as in the case of printed documents and/or removable media. Explain how the use of those attributes in the destination environment will achieve part of the necessary security protection that caused the inclusion of the related objective.

**O.Data\_Imp\_Exp\_Control:** Data import/export to/from system control  
Protect data from being sent to erroneous places and more places external to the system than allowed by the organization's security policy. Conversely the import of data into the system should be protected from illicit information or information not allowed by the organization's security policy.

O.Data\_Imp\_Exp\_Control is implemented in the TOE by:

1. FDP\_ETC.2: Export of user data with security attributes

**Component Application Rationale:** Provide export of user data from the TOE with security attributes that limit the number of locations and specify which locations that the data can be sent to.

2. FDP\_IFF.1: Simple security attributes

**Component Application Rationale:** A user could send unauthorized information. Therefore limits should be placed on the types of acceptable information flows and the limits on information flows. This would reduce bandwidth utilization and enforce organizational policy.

3. FDP\_ITC.2: Import of user data with security attributes

**Component Application Rationale:** Provide import of user data from the TOE with security attributes that limit the locations that a user can accept data from.

**O.EMSEC\_Design:** Provide physical emanations security  
Design and build the system in such a way as to control the production of intelligible emanations within specified limits.

**Implementation Application:** \$

Support with a CC-Extending component along the following lines:

Name. FPT\_PHP\_EMSEC\_Design, Physical Emanations Security

Overview. IT equipment that processes sensitive user data may need to avoid the unintended transmission of information-bearing sounds or electromagnetic signals.

Justification of explicit statement. FDP\_UCT.1.1 is similar, but is clearly aimed at

deliberate communications, appearing not to cover unintended emanations. FPT\_PHP claims generally to deal with physical protection, but the individual components all address physical modification rather than eavesdropping.

Applicability of Assurance Requirements. Special skills may well be needed to evaluate satisfaction of this requirement. However, there are commercial and government testing laboratories that could be consulted.

FPT\_PHP\_EMSEC\_D.1. The TOE shall not emit [assignment: types of emissions] in excess of [assignment: specified limits].

FPT\_PHP\_EMSEC\_D.2. The TOE connecting cables (power supply, communications lines) shall not emit [assignment: types of emissions] in excess of [assignment: specified limits].

Types of emanations: intelligible

Limits: sufficient to mitigate intelligible emanations when used in conjunction with other objectives.

Objective Rationale. \$

O.EMSEC\_Design is implemented in the TOE by:

1. ADV\_FSP.1: Informal functional specification
2. ADV\_HLD.2: Security enforcing high-level design
3. ATE\_COV.2: Analysis of coverage
4. ATE\_DPT.1: Testing: high-level design
5. ATE\_FUN.1: Functional testing
6. ATE\_IND.2: Independent testing - sample
7. FPT\_PHP\_EMSEC\_Design: Physical Emanations Security

O.Export\_Control: Sanitize data objects containing hidden or unused data

Sanitize data objects that may contain hidden data when they are exported from the TOE in order to inhibit steganographic smuggling.

O.Export\_Control is implemented in the TOE by:

1. FDP\_ACC.2: Complete access control

O.External\_Labels: Label or mark information for external systems

Label or mark information for external systems to prevent the exchange of inappropriate data between systems.

O.External\_Labels is implemented in the TOE by:

1. FDP\_ETC.2: Export of user data with security attributes
2. FDP\_ITC.2: Import of user data with security attributes

O.Fail\_Secure: Preservation of secure state for failures in critical components  
Preserve the secure state of the system in the event of a secure component failure.

Implementation Application: Choose FPT\_FLS.1 to implement this objective.

---

This objective applies to the secure state of the system in its entirety.

O.Fail\_Secure is implemented in the TOE by:

1. FPT\_FLS.1: Failure with preservation of secure state

Component Application: To support the implementation of this requirement, the developer must provide a definition of "secure state" so that the requirement can be evaluated. This definition can be provided through the associated dependency on ADV\_SPM.1.

Component Application Rationale: The definition of a "secure state" should be provided by the security model documentation (ADV\_SPM.1) when a model is required by the PP. If the requirement for a security model is not present, it is still necessary for the developer to provide this definition in some manner. How this definition is provided should be identified in the rationale that defends the exclusion of ADV\_SPM.1.

O.Fault\_Tolerance: Provide fault tolerant operations for critical components  
Provide fault tolerant operations for critical components and continue to operate in the presence of specific failures in one or more system components.

Implementation Application: There are two variants of this objective (choose one):

[Basic]: Choose FRU\_FLT.1

[Resistant]: Choose FRU\_FLT.2

Choose the [Basic] variant if only some system capabilities need to be fault tolerant to some failures.

Choose the [Resistant] variant if all system capabilities need to be fault tolerant to some failures. In general, the [Resistant] variant is much more difficult to implement than the [Basic] variant.

---

However, the TSF must guarantee that tolerance (e.g., continuing operation) of the TSF in the presence of the identified failures does not at the expense of the TSF's secure state (see FPT\_FLS.1). The types of failures and the related TOE capabilities listed in this component should be strongly correlated with the resultant risk, otherwise the degree of protection provided may be inadequate or not cost effective.

O.Fault\_Tolerance is implemented in the TOE by:

1. FRU\_FLT.1: Degraded fault tolerance

Component Application Rationale: The objective expresses the CC component in narrative form.

O.General\_Integ\_Checks: Periodically check integrity  
Provide periodic integrity checks on both system and user data.

O.General\_Integ\_Checks is implemented in the TOE by:

1. FPT\_TST.1: TSF testing
2. FDP\_SDI.2: Stored data integrity monitoring and action

O.Hack\_Limit\_Sessions: Limit sessions to outside users  
Limit the number of sessions available to outside users. A hacker can initiate multiple communication sessions that could cause an overload on resources, for example, half open session starts as is seen in "SYN flood" attacks.

O.Hack\_Limit\_Sessions is implemented in the TOE by:

1. AGD\_ADM.1: Administrator guidance

Component Application: Include guidance for the security administrator to reflect the organization's policy for limiting hacker sessions. For example, this guidance would give the administrator information for setting thresholds to counter a "SYN flood" attack.

2. FMT\_MSA.1: Management of security attributes

Component Application: Apply this component to SFP's dealing with communications ports. For example, the TSF may prevent creation of ports or redirect suspected hacker traffic to a harmless destination for analysis.

O.Info\_Flow\_Control: System enforced information flow  
Enforce an information flow policy whereby users are constrained from allowing access to information they control, regardless of their intent (e.g., mandatory access control). This lattice property of security attributes is commonly associated with the U.S. DoD implementations of Mandatory Access Control (MAC).

Implementation Application: Four variants of this objective exist (choose one).  
[Subset-Simple] Choose FDP\_IFC.1 and FDP\_IFF.1.  
[Subset-Lattice] Choose FDP\_IFC.1 and FDP\_IFF.2.  
[Complete-Simple] Choose FDP\_IFC.2 and FDP\_IFF.1.  
[Complete-Lattice] Choose FDP\_IFC.2 and FDP\_IFF.2.



Each of the four variants has an FDP\_IFC component to define the policy and an FDP\_IFF component to describe how the policy is to be enforced. FDP\_IFC.1 and FDP\_IFC.2 differ in the scope of the information flow policy. FDP\_IFF.1 and FDP\_IFF.2 differ in the scope of the information flow policy and the structure of the security attributes used. In the latter case, the attribute values must be partially-ordered in a relationship called a "lattice."

O.Info\_Flow\_Control is implemented in the TOE by:

1. FDP\_IFC.2: Complete information flow control
2. FDP\_IFF.1: Simple security attributes

O.Input\_Inspection: Require inspection for absence of malicious code.  
Require inspection of downloads/transfers.

O.Input\_Inspection is implemented in the TOE by:

1. FDP\_ACF.1: Security attribute based access control

Component Application: Use this component to specify downloaded executables as a class of objects to be covered by security function requirements.

2. FDP\_ITC.1: Import of user data without security attributes

Component Application: Provide an importation control rule requiring inspection of downloads.

3. FDP\_ACC.2: Complete access control

O.Integ\_Data\_Mark\_Exp: Data marking integrity export  
Ensure that data markings are included with data that is exported to another trusted product.

O.Integ\_Data\_Mark\_Exp is implemented in the TOE by:

1. FDP\_ETC.2: Export of user data with security attributes

O.Integ\_Sys\_Data\_Ext: Integrity of system data transferred externally  
Ensure the integrity of system data exchanged externally with another trusted product by using a protocol for data transfer that will permit error detection and correction.

This includes detecting and possibly correcting errors in data received and encoding outgoing data to make it possible for the receiver to detect and possibly correct errors. The method for detecting and correcting errors is based on some method (protocol) that is agreed upon by participating parties.

O.Integ\_Sys\_Data\_Ext is implemented in the TOE by:

1. FPT\_ITL.1: Inter-TSF detection of modification

O.Integ\_Sys\_Data\_Int: Integrity of system data transferred internally  
Ensure the integrity of system data transferred internally.

Implementation Application: Two variants of this objective can be implemented:  
[CASE-A] Choose FPT\_ITT.1 and FPT\_SSP.1.  
[CASE-B] Choose FPT\_ITT.1 and FPT\_SSP.2.

FPT\_SSP.1: This component is recommended when the system is distributed. This requirement provides simple acknowledgement by the data recipient.

FPT\_SSP.2: This component is recommended when the system is distributed. This requirement provides mutual acknowledgement of the data exchange.

O.Integ\_Sys\_Data\_Int is implemented in the TOE by:

1. FPT\_SSP.2: Mutual trusted acknowledgement
2. FPT\_ITT.2: TSF data transfer separation

O.Integ\_User\_Data\_Int: Protect user data during internal transfer  
Ensure the integrity of user data transferred internally within the system.

O.Integ\_User\_Data\_Int is implemented in the TOE by:

1. FDP\_ITT.2: Transmission separation by attribute

O.Integrity\_Data/SW: Integrity protection for user data and software  
Provide integrity protection for user data and software.

O.Integrity\_Data/SW is implemented in the TOE by:

1. FDP\_SDI.2: Stored data integrity monitoring and action

O.Integrity\_Data\_Rep: Integrity of system data replication  
Ensure that when system data replication occurs across the system the data is consistent for each replication.

O.Integrity\_Data\_Rep is implemented in the TOE by:

1. FPT\_TRC.1: Internal TSF consistency

Component Application Rationale: The objective expresses the CC component in narrative form.

O.Integrity\_Practice: Operational integrity system function testing  
Provide system functional tests to periodically test the integrity of the hardware and code running system functions.

O.Integrity\_Practice is implemented in the TOE by:

1. FPT\_AMT.1: Abstract machine testing
2. FPT\_TST.1: TSF testing

O.IntelEman\_Contain: Emanations containment  
Confine system-produced intelligible emanations to within a specified limit.

Implementation Application: This objective should be allocated to the TOE environment.

Support with a CC-Extending component along the following lines:

Name. F\_PhysEnv\_Cnf.1, Emanations Security

Overview. The physical TOE environment shall provide an effective container that prevents the escape of intelligible electromagnetic and sound vibrations generated by the TOE.

Justification of explicit statement. This is explicitly an environmental requirement, in contrast to existing Part 2 CC components.

Applicability of Assurance Requirements. Special skills may well be needed to evaluate satisfaction of this requirement. However, there are commercial and government testing laboratories that could be consulted.

F\_PhysEnv\_Cnf.1.1. The physical TOE environment shall not permit the escape of [assignment: type of TOE emissions] in excess of [assignment: specified limit].

Type of emissions: intelligible

Limits: sufficient to mitigate intelligible emanations when used in conjunction with other objectives.

Objective Rationale. \$

O.IntelEman\_Contain is implemented in the TOE by:

1. ATE\_IND.2: Independent testing - sample
2. ADV\_HLD.2: Security enforcing high-level design
3. ADV\_FSP.1: Informal functional specification

O.IntelEman\_Control: Emanations control

Limit system-produced intelligible emanations to within a specified limit.

Implementation Application: \$

Support with a CC-Extending component along the following lines:

Name. FPT\_PHP\_EMSEC\_Design, Physical Emanations Security

Overview. IT equipment that processes sensitive user data may need to avoid the unintended transmission of information-bearing sounds or electromagnetic signals.

Justification of explicit statement. FDP\_UCT.1.1 is similar, but is clearly aimed at deliberate communications, appearing not to cover unintended emanations. FPT\_PHP claims generally to deal with physical protection, but the individual components all address physical modification rather than eavesdropping.

Applicability of Assurance Requirements. Special skills may well be needed to evaluate satisfaction of this requirement. However, there are commercial and government testing laboratories that could be consulted.

FPT\_PHP\_EMSEC\_D.1. The TOE shall not emit [assignment: types of emissions] in excess of [assignment: specified limits].

FPT\_PHP\_EMSEC\_D.2. The TOE connecting cables (power supply, communications lines) shall not emit [assignment: types of emissions] in excess of [assignment: specified limits].

Types of emissions: intelligible

Limits: sufficient to mitigate intelligible emanations when used in conjunction with other objectives.

Objective Rationale. \$

O.IntelEman\_Control is implemented in the TOE by:

1. ATE\_IND.2: Independent testing - sample
2. ADV\_FSP.1: Informal functional specification
3. ADV\_HLD.2: Security enforcing high-level design

O.Lifecycle\_Security: Lifecycle security

Provide tools, techniques, and security employed during the development phase. Detect and resolve flaws during the operational phase. Provide safe destruction techniques.

O.Lifecycle\_Security is implemented in the TOE by:

1. ALC\_DVS.1: Identification of security measures

2. ALC\_LCD.1: Developer defined life-cycle model

O.Limit\_Actions\_Auth: Restrict actions before authentication

Restrict the actions a user may perform before the TOE verifies the identity of the user.

O.Limit\_Actions\_Auth is implemented in the TOE by:

1. FIA\_UAU.2: User authentication before any action

O.Limit\_Comm\_Sessions: Limit the number of user initiated communication sessions

Provide mechanisms to limit the number of sessions that the user can initiate, if the user initiates multiple sessions that exceed the processors ability to perform in a reliable and efficient manner. These sessions could either be communication (TCP/IP) sessions or user login sessions.

O.Limit\_Comm\_Sessions is implemented in the TOE by:

1. FTA\_MCS.1: Basic limitation on multiple concurrent sessions

Component Application: The system should limit the number of multiple sessions to the amount that is expected to maximize any resource utilization.

Component Application Rationale: The resource that the threat agent is attacking should be the primary metric for the maximum number of multiple sessions.

O.Maintain\_Sec\_Domain: Maintain security domain

Maintain at least one security domain for system (TOE) execution to protect the TOE from interference and tampering.

O.Maintain\_Sec\_Domain is implemented in the TOE by:

1. FPT\_SEP.2: SFP domain separation

O.Malicious\_Code: Procedures for preventing malicious code

Incorporate malicious code prevention procedures and mechanisms.

O.Malicious\_Code is implemented in the TOE by:

1. FDP\_ITC.1: Import of user data without security attributes

2. FPT\_AMT.1: Abstract machine testing
3. FPT\_PHP.1: Passive detection of physical attack
4. FPT\_TST.1: TSF testing

O.Manage\_Res\_Sec\_Attr: Manage resource security attributes  
Provide management on resource security attributes.

O.Manage\_Res\_Sec\_Attr is implemented in the TOE by:

1. AGD\_USR.1: User guidance

Component Application: Provide guidance to the user that would define the level of security attribute modification they are allowed and what the ramifications are for violating that policy.

2. FAU\_GEN.1: Audit data generation

Component Application: Level of auditing: This should generally not be determined on the basis of a single application of this component, but through consideration of audit requirements in their entirety.

Maintain an audit trail of resource security attribute modifications or additions.

Component Application Rationale: Maintain an audit trail of resource security attribute modifications or additions.

3. FMT\_MSA.1: Management of security attributes

O.Manage\_TSF\_Data: Manage security-critical data to avoid storage space being exceeded

Manage security-critical (TSF) data to ensure that the size of the data does not exceed the space allocated for storage of the data.

O.Manage\_TSF\_Data is implemented in the TOE by:

1. FMT\_MTD.2: Management of limits on TSF data

Component Application: FMT\_MTD.2.1/If data storage can be exceeded, the roles of who can modify limits on TSF data should be defined.

FMT\_MTD2.2/If data storage can be exceeded, the actions required upon storage limits being exceeded should be defined.

**O.MessageAuthentication:** Message authentication  
Each requestor must prove knowledge of the shared secret.

O.MessageAuthentication is implemented in the TOE by:

1. FCO\_NRO.1: Selective proof of origin
2. FCO\_NRO.2: Enforced proof of origin
3. FCO\_NRR.1: Selective proof of receipt
4. FCO\_NRR.2: Enforced proof of receipt
5. FCS\_COP.1: Cryptographic operation
6. FIA\_AFL.1: Authentication failure handling
7. FIA\_UAU.2: User authentication before any action
8. FIA\_UAU.4: Single-use authentication mechanisms
9. FIA\_UID.2: User identification before any action
10. FDP\_ACC.2: Complete access control
11. FDP\_ACF.1: Security attribute based access control
12. FDP\_DAU.1: Basic data authentication
13. FPT\_RPL.1: Replay detection

**O.MetricReporting:** Integrity metric reporting  
The TOE must report the values in the current PCR registers. The report may be digitally signed.

O.MetricReporting is implemented in the TOE by:

1. ADV\_HLD.2: Security enforcing high-level design
2. ADV\_FSP.1: Informal functional specification

**O.NoBore:** No BORE attacks  
The TOE provides protection from Break Once Run Everywhere attacks.

O.NoBore is implemented in the TOE by:

1. ADV\_FSP.1: Informal functional specification
2. ADV\_HLD.2: Security enforcing high-level design
3. ADV\_SPM.1: Informal TOE security policy model

**O.No\_Residual\_Info:** Eliminate residual information  
Ensure there is no "object reuse;" i.e., ensure that there is no residual information in some information containers or system resources upon their reallocation to different users.

**Implementation Application:** Select one of FDP\_RIP.1 or FDP\_RIP.2 to specify the scope of protection against disclosure of residual information.

O.No\_Residual\_Info is implemented in the TOE by:

1. FDP\_RIP.1: Subset residual information protection
2. FDP\_RIP.2: Full residual information protection

O.NonRepud\_Assess\_Recd: Non-repudiation support for received information by a nonlocal sender's TSF

Support nonrepudiation for received information by supporting remote handling of nonrepudiation evidence if needed.

O.NonRepud\_Assess\_Recd is implemented in the TOE by:

1. AGD\_ADM.1: Administrator guidance

Component Application: In the event that remote third parties are involved in enforcing nonrepudiation, provide guidance on proper use of the nonrepudiation functions.

Component Application Rationale: In the event that a third party is responsible for marshalling gathered evidence to enforce accountability, there needs to be a requirement for performing this duty.

2. AGD\_USR.1: User guidance

Component Application: In the event that senders are involved in enforcing nonrepudiation, provide guidance on proper use of the nonrepudiation functions.

Component Application Rationale: In the event that a senders are responsible for marshalling gathered evidence to enforce accountability, there needs to be user documentation on how to do this.

3. FCO\_NRR.1: Selective proof of receipt

Component Application: Information types: possibly none, as evidence is generated by the receiving TSF.

Requestors, recipients of evidence: specify originator, recipient, or third parties.

Attributes, Information fields: those supported by the receiving TSF.

Limitations on evidence of receipt: TSF must rely on receiving TSF to supply evidence of receipt.

Component Application Rationale: The TOE must rely on the recipient's TSF to generate evidence, which is then assessed by the TOE in light of any additional information available.



4. FMT\_MOF.1: Management of security functions behaviour

Component Application: In the case where nonlocal third parties are involved in handling nonrepudiation evidence, provide identified roles for those parties.

Component Application Rationale: In the event that a third party is responsible for marshalling gathered evidence to enforce accountability, this requirement provides appropriate role support.

O.NonRepud\_Assess\_Sent: Non-repudiation support for sent information by the nonlocal receiving TSF.

Support nonrepudiation for sent information by supporting remote handling of nonrepudiation evidence if needed.

O.NonRepud\_Assess\_Sent is implemented in the TOE by:

1. AGD\_ADM.1: Administrator guidance

Component Application: In the event that nonrepudiation evidence is handled by remote third parties, provide documentation on the responsibilities of those who handle this evidence.

Component Application Rationale: In the event that a third party is responsible for marshalling gathered evidence to enforce accountability, there needs to be a requirement for performing this duty.

2. AGD\_USR.1: User guidance

Component Application: In the event that nonrepudiation evidence is handled by the message recipient, provide user documentation on the use of nonrepudiation features.

Component Application Rationale: In the event that recipients are responsible for marshalling gathered evidence to enforce accountability, there needs to be user documentation on how to do this.

3. FCO\_NRO.1: Selective proof of origin

Component Application: Transmitted information types: possibly none, as the TOE must rely on the sending TSF for generated evidence.

Requestors, Recipients of evidence: specify originator, recipient, or third parties.

Attributes, information fields: those supported by the sending TSF.

Limitations on the evidence of origin: The TOE must rely on the sending TSF for evidence of message generation.

Component Application Rationale: The TOE must rely on the sender's TSF to

generate evidence, which is then assessed by the TOE in light of any additional information available.

4. FMT\_MOF.1: Management of security functions behaviour

Component Application: If nonrepudiation is handled by remote third parties, apply FMT\_MOF.1 Management of security functions behavior to provide identified roles for the parties who determine the use of nonrepudiation functions.

Component Application Rationale: In the event that a third party is responsible for marshalling gathered evidence to enforce accountability, this requirement provides appropriate role support.

O.NonRepud\_Gen\_Recd: Non-repudiation support for received information by the recipient's TSF

Prevent a receiving user from avoiding accountability for receiving a message by providing evidence that the user received the message.

Implementation Application: The objective has multiple implementations. Using FCO\_NRR.2 is a special case of using FCO\_NRR.1 and may be stronger. FMT\_MOF.1 and AGD\_ADM.1 are needed if threat countering is to be performed by third parties using the same system as the recipient.

O.NonRepud\_Gen\_Recd is implemented in the TOE by:

1. AGD\_ADM.1: Administrator guidance

Component Application: In the event that third parties are involved in enforcing nonrepudiation, provide guidance on proper use of the nonrepudiation functions.

Component Application Rationale: In the event that a third party is responsible for marshalling gathered evidence to enforce accountability, there needs to be a user requirement for performing this duty.

2. FCO\_NRR.1: Selective proof of receipt

Component Application: List the information types for which the threat is significant, the parties responsible for countering the threat (typically the sender or a trusted third party), the attributes of the recipient that are subject to nonrepudiation (e.g., individual identity, employing organization), information fields subject to nonrepudiation (e.g., message headers and text), parties to whom nonrepudiation evidence is presented (e.g., those responsible for countering the attack), and any limitations on the evidence provided.

Component Application Rationale: This requirement ensures that nonrepudiation evidence can be provided.

3. FCO\_NRR.2: Enforced proof of receipt

Component Application: List the information types for which the threat is significant, the parties responsible for countering the threat (typically the sender or a trusted third party), the attributes of the recipient that are subject to nonrepudiation (e.g., individual identity, employing organization), information fields subject to nonrepudiation (e.g., message headers and text), parties to whom nonrepudiation evidence is presented (e.g., those responsible for countering the attack), and any limitations on the evidence provided.

Component Application Rationale: This requirement ensures that evidence of nonrepudiation is provided.

4. FMT\_MOF.1: Management of security functions behaviour

Component Application: In the event that third parties are involved in enforcing nonrepudiation, provide identified roles for these parties.

Component Application Rationale: In the event that a third party is responsible for marshalling gathered evidence to enforce accountability, this requirement provides appropriate role support.

O.NonRepud\_Gen\_Sent: Non-repudiation support for sent information by the sender's TSF.

Prevent a user from avoiding accountability for sending a message to a recipient at a different site by providing evidence that the user sent the message.

Implementation Application: The objective has multiple implementations. Using FCO\_NRO.2 is a special case of using FCO\_NRO.1 and may be stronger. FMT\_MOF.1 and AGD\_ADM.1 are needed only if threat countering is to be performed by third parties (i.e., people other than the recipient).

O.NonRepud\_Gen\_Sent is implemented in the TOE by:

1. AGD\_ADM.1: Administrator guidance

Component Application: In the event that nonrepudiation evidence is handled by third parties at the sender's system, provide documentation on the responsibilities of those who handle this evidence.

Component Application Rationale: In the event that a third party is responsible for marshalling gathered evidence to enforce accountability, there needs to be a requirement for performing this duty.

2. FCO\_NRO.1: Selective proof of origin

Component Application: List the information types for which the threat is significant, parties responsible for countering the threat (typically the recipient or a trusted third party), the attributes of the originator that are subject to nonrepudiation (e.g., individual identity, employing organization), information fields subject to nonrepudiation (some fields may be added by the TOE and not under the sender's control), parties to whom nonrepudiation evidence is presented (e.g., those responsible for countering the attack), and any limitations on the evidence provided.

Component Application Rationale: This requirement ensures that nonrepudiation evidence can be provided.

3. FCO\_NRO.2: Enforced proof of origin

Component Application: List the information types for which the threat is significant, parties responsible for countering the threat (typically the recipient or a trusted third party), the attributes of the originator that are subject to nonrepudiation (e.g., individual identity, employing organization), information fields subject to nonrepudiation (some fields may be added by the TOE and not under the sender's control), parties to whom nonrepudiation evidence is presented (e.g., those responsible for countering the attack), and any limitations on the evidence provided.

Component Application Rationale: This requirement ensures that nonrepudiation evidence is generated.

4. FMT\_MOF.1: Management of security functions behaviour

Component Application: If nonrepudiation is handled by third parties at the sender's system, apply FMT\_MOF.1 Management of security functions behavior to provide identified roles for the parties who use the nonrepudiation functions.

Component Application Rationale: In the event that a third party is responsible for marshalling gathered evidence to enforce accountability, this requirement provides appropriate role support.

O.Obj\_Attr\_Integrity: Basic object attribute integrity  
Maintain object security attributes with moderate to high accuracy (under the guidance of qualified users).

O.Obj\_Attr\_Integrity is implemented in the TOE by:

1. FDP\_ACF.1: Security attribute based access control

Component Application: Access control SFP: attribute management

Security attributes: object owner (a.k.a. object-attribute manager)

Rules governing access: only an object owner may apply attribute-maintenance operations to that object, and only via attribute-maintenance subjects acting on that user's behalf; all attribute-maintenance subjects are certified to reflect user intent when modifying object attributes.

2. FMT\_MSA.1: Management of security attributes

Component Application: Role: data security administrator.  
Ability: change\_default, modify, delete attributes.

3. FMT\_MSA.2: Secure security attributes

4. FMT\_MSA.3: Static attribute initialisation

Component Application: Policy: attribute-management.  
Default values: safe values - determined separately for each attribute per its associated SFP.  
Roles authorized to specify initial values for object data and attributes: TOE developer and/or the TOE data security administrator.

5. FDP\_ACC.2: Complete access control

6. FMT\_SMR.2: Restrictions on security roles

O.Obj\_Protection: Object domain protection

Require domain protection for objects. Specify object classes (domains), user groups, and operation classes. Use these to specify which operations may be performed on which objects by which users. Basically this controls what users can do in a given group.

O.Obj\_Protection is implemented in the TOE by:

1. FDP\_ACF.1: Security attribute based access control

Component Application: Limit ability to modify trusted objects to a trusted role

2. FMT\_MSA.3: Static attribute initialisation

Component Application: Create a trusted role for the maintenance of trusted objects.

3. FDP\_ACC.2: Complete access control

O.Prevent\_Link: Prevent linking of multiple service use  
Ensure that a user may make multiple uses of a service or resource without other specified users being able to link these uses together.

Implementation Application: Choose FPR\_UNL.1 to implement the objective.

O.Prevent\_Link is implemented in the TOE by:

1. FPR\_UNL.1: Unlinkability

Component Application Rationale: The objective expresses the CC component in narrative form.

O.Protected\_Capability: Protected Capability and shielded location  
The TOE must identify and protect capabilities as defined in the TCPA specification.

O.Protected\_Capability is implemented in the TOE by:

1. ADV\_HLD.2: Security enforcing high-level design
2. ADV\_FSP.1: Informal functional specification
3. FCS\_COP.1: Cryptographic operation
4. FCS\_CKM.1: Cryptographic key generation
5. FCS\_CKM.4: Cryptographic key destruction
6. FDP\_ITT.4: Attribute-based integrity monitoring

O.Rcv\_MsgMod\_ID: Identify message modification in messages received  
The TSF recognizes changes to messages that occurred in transit, including insertion of spurious messages and deletion or replay of legitimate messages.

O.Rcv\_MsgMod\_ID is implemented in the TOE by:

1. FDP\_UTI.1: Data exchange integrity

Component Application: TOE: Specify received  
Environment: Specify sent

O.React\_Discovered\_Atk: React to discovered attacks  
Implement automated notification or other reactions to the TSF-discovered attacks in an effort to identify attacks and to create an attack deterrent.

O.React\_Discovered\_Atk is implemented in the TOE by:

1. AGD\_ADM.1: Administrator guidance
2. FAU\_ARP.1: Security alarms

O.RootMeasurement: Measurement root of trust

The entity that provides the base for measuring integrity values is the measurement root of trust. This entity on a PC would be the boot block or something similar.

O.RootMeasurement is implemented in the TOE by:

1. ADV\_FSP.1: Informal functional specification
2. ADV\_HLD.2: Security enforcing high-level design

O.RootReporting: Reporting root of trust

The reporting root of trust is the endorsement key. This provides a singular point that all challengers can rely on.

O.RootReporting is implemented in the TOE by:

1. ADV\_FSP.1: Informal functional specification
2. ADV\_HLD.2: Security enforcing high-level design
3. ADV\_SPM.1: Informal TOE security policy model

O.SecureManufacturing: Secure TPM creation and certification

The TPM manufacturing process requires the creation and certification of the endorsement key. The TPM manufacturing process must perform this creation and certification in a manner that provides the assurances that the endorsement key was properly created. The process must also provide assurances that the certification of the endorsement key is done with the correct private key and that the process protects the certification key and properly protects certification process.

Implementation Application: The process of creating a TPM and certifying the endorsement key are crucial to the "trust" in the TPM. The TPM manufacturer must provide this trust by having a manufacturing process that can show the proper care of both the TPM and its endorsement key and the protection of the private key that performs the certification.

O.SecureManufacturing is implemented in the TOE by:

1. ALC\_DVS.1: Identification of security measures

O.Secure\_State: Protect and maintain secure system state

Maintain and recover to a secure state without security compromise after system error or

other interruption of system operation.

**Implementation Application:** In applying this objective, provide a secure state definition that captures the notion of being able to enforce the TSP. If the TSP isn't conveniently described in terms of a secure-state model, the PP author may wish to introduce a CC-extending component that recasts FPT\_RCV to accommodate the PP's TSP description.

O.Secure\_State is implemented in the TOE by:

1. FPT\_FLS.1: Failure with preservation of secure state
2. FPT\_RCV.3: Automated recovery without undue loss
3. FPT\_RCV.4: Function recovery

**O.Security\_Attr\_Mgt:** Manage security attributes  
Manage the initialization of, values for, and allowable operations on security attributes.

**Implementation Application:** Three variants of this objective exist (choose any):

[Restrict]: Choose FMT\_MSA.1.  
[Secure]: Choose FMT\_MSA.2.  
[Init]: Choose FMT\_MSA.3.

O.Security\_Attr\_Mgt is implemented in the TOE by:

1. FMT\_MSA.1: Management of security attributes
2. FMT\_MSA.2: Secure security attributes
3. FMT\_MSA.3: Static attribute initialisation

**O.Security\_Data\_Mgt:** Manage security-critical data  
Manage the initialization of, limits on, and allowable operations on security-critical data.

**Implementation Application:** Three variants of this objective exist (choose any):

[Restrict]: Choose FMT\_MTD.1.  
[Limits]: Choose FMT\_MTD.2.  
[Secure]: Choose FMT\_MTD.3.

O.Security\_Data\_Mgt is implemented in the TOE by:



1. FMT\_MTD.1: Management of TSF data
2. FMT\_MTD.2: Management of limits on TSF data
3. FMT\_MTD.3: Secure TSF data

O.Security\_Func\_Mgt: Manage behavior of security functions  
Provide management mechanisms for security mechanisms.

Implementation Application: Choose FMT\_MOF.1 to implement this objective.

O.Security\_Func\_Mgt is implemented in the TOE by:

1. FMT\_MOF.1: Management of security functions behaviour

O.Security\_Roles: Security roles  
Maintain security-relevant roles and the association of users with those roles.

Implementation Application: Two variants of this objective exist.

[Basic] Choose FMT\_SMR.1. This variation provides the basic capability to define different roles among users of the system.

[Restricted] Choose FMT\_SMR.2. This variation provides the additional capability of placing constraints on use of the defined roles within the system.

O.Security\_Roles is implemented in the TOE by:

1. FMT\_SMR.2: Restrictions on security roles

O.Snt\_MsgMod\_ID: Identify message modification in messages sent  
The TSF supports recognition of changes to transmitted messages that occurred in transit, including insertion of spurious messages and deletion or replay of legitimate messages.

O.Snt\_MsgMod\_ID is implemented in the TOE by:

1. FDP\_UIT.1: Data exchange integrity

Component Application: TOE: Specify received  
Environment: Specify sent

O.Source\_Code\_Exam: Examine the source code for developer flaws  
Examine for accidental or deliberate flaws in code made by the developer. The accidental

flaws could be lack of engineering detail or bad design. Where the deliberate flaws would include building trapdoors for later entry as an example.

O.Source\_Code\_Exam is implemented in the TOE by:

1. ADV\_RCR.1: Informal correspondence demonstration

Component Application Rationale: To be used as a dependency for ADV\_IMP.2 in O.Source\_Code\_Exam.

O.SpecRef: Specification reference

The TOE must provide all of the features and functions as specified in the TCPA specification.

O.SpecRef is implemented in the TOE by:

1. FCS\_CKM.1: Cryptographic key generation
2. FCS\_CKM.4: Cryptographic key destruction
3. FPT\_ITT.2: TSF data transfer separation
4. FPT\_PHP.1: Passive detection of physical attack
5. FPT\_PHP.3: Resistance to physical attack
6. FPT\_RPL.1: Replay detection
7. ACM\_SCP.1: TOE CM coverage
8. ADO\_DEL.1: Delivery procedures
9. ADO\_IGS.1: Installation, generation, and start-up procedures
10. ADV\_FSP.1: Informal functional specification
11. ADV\_HLD.2: Security enforcing high-level design
12. ADV\_RCR.1: Informal correspondence demonstration
13. ADV\_SPM.1: Informal TOE security policy model
14. AGD\_ADM.1: Administrator guidance
15. AGD\_USR.1: User guidance
16. ALC\_DVS.1: Identification of security measures
17. ALC\_LCD.1: Developer defined life-cycle model
18. AVA\_MSU.1: Examination of guidance
19. AVA\_SOF.1: Strength of TOE security function evaluation
20. AVA\_VLA.1: Developer vulnerability analysis
21. ATE\_DPT.1: Testing: high-level design
22. ATE\_FUN.1: Functional testing
23. ATE\_IND.2: Independent testing - sample
24. ACM\_CAP.3: Authorisation controls
25. ATE\_COV.2: Analysis of coverage
26. FDP\_ITT.4: Attribute-based integrity monitoring
27. FPT\_TDC.1: Inter-TSF basic TSF data consistency
28. FRU\_PRS.1: Limited priority of service

O.Standard\_Output\_Pres: Standard presentation of output values  
Present each possible output value in a standard form.

O.Standard\_Output\_Pres is implemented in the TOE by:

1. ADV\_FSP.1: Informal functional specification
2. ADV\_HLD.2: Security enforcing high-level design

O.Storage\_Integrity: Storage integrity  
Provide integrity for data.

O.Storage\_Integrity is implemented in the TOE by:

1. FDP\_SDI.2: Stored data integrity monitoring and action

O.Sys\_Assur\_HW/SW/FW: Validation of security function  
Ensure that security-relevant software, hardware, and firmware are correctly functioning through features and procedures.

O.Sys\_Assur\_HW/SW/FW is implemented in the TOE by:

1. ATE\_FUN.1: Functional testing
2. FPT\_TST.1: TSF testing

O.Sys\_Backup\_Procs: System backup procedures  
Provide backup procedures to ensure that the system can be reconstructed.

O.Sys\_Backup\_Procs is implemented in the TOE by:

1. FPT\_RCV.3: Automated recovery without undue loss

O.Sys\_Backup\_Verify: Detect modifications of backup hardware, firmware, software  
Detect modifications to backup hardware, firmware, and software.

Implementation Application: The component FPT\_PHP.1 would be used to provide the physical protection of the backup hardware, firmware, and software. The components FPT\_TST.1 and FPT\_AMT.1 would be used to test whether the secure state of the backup hardware, firmware, and software is maintained and restored if necessary.

O.Sys\_Backup\_Verify is implemented in the TOE by:

1. FPT\_AMT.1: Abstract machine testing
2. FPT\_PHP.1: Passive detection of physical attack
3. FPT\_TST.1: TSF testing

O.Sys\_Self\_Protection: Protection of system security function  
Protect the system security functions through technical features.

O.Sys\_Self\_Protection is implemented in the TOE by:

1. FPT\_SEP.2: SFP domain separation

O.TCPAIdentities: TCPA Identities  
The TOE must provide the ability to create, manage and use identities.

O.TCPAIdentities is implemented in the TOE by:

1. FCS\_CKM.1: Cryptographic key generation
2. FCS\_CKM.4: Cryptographic key destruction

O.TCPAProtectedStorage: TCPA Protective Storage  
The TOE must provide a protected storage mechanism as defined in the specification section 3.6 and chapter 6.

O.TCPAProtectedStorage is implemented in the TOE by:

1. FDP\_ACC.2: Complete access control
2. FDP\_ACF.1: Security attribute based access control
3. FDP\_DAU.1: Basic data authentication
4. FDP\_ETC.2: Export of user data with security attributes
5. FDP\_IFC.2: Complete information flow control
6. FDP\_IFF.1: Simple security attributes
7. FDP\_ITC.1: Import of user data without security attributes
8. FDP\_ITC.2: Import of user data with security attributes
9. FDP\_ITT.2: Transmission separation by attribute
10. FDP\_RIP.1: Subset residual information protection
11. FDP\_RIP.2: Full residual information protection
12. FDP\_ROL.1: Basic rollback
13. FDP\_SDI.2: Stored data integrity monitoring and action
14. FDP\_UCT.1: Basic data exchange confidentiality
15. FDP\_UIT.1: Data exchange integrity
16. FIA\_AFL.1: Authentication failure handling
17. FIA\_UAU.2: User authentication before any action

18. FPR\_ANO.2: Anonymity without soliciting information

O.TSF\_Rcv\_Err\_ID\_Loc: Local detection of received security-critical data modified in transit

Identification by the system (TOE) of modification of security-critical (TSF) data occurring in transit from a remote trusted site must occur.

O.TSF\_Rcv\_Err\_ID\_Loc is implemented in the TOE by:

1. FPT\_ITL.1: Inter-TSF detection of modification

Component Application: For the TOE: Modification metric: specify desired strength of function for received messages (must be compatible with that used by the remote site).

Action if modification of imported data is detected: Specify any desired responses made by the TSF.

For the IT Environment: Modification metric: specify desired strength of function for exported data. (The modification metric for sent TSF data must be compatible with that used by the TOE.)

Action if modification of exported data is detected: none necessary.

O.TSF\_Rcv\_Err\_ID\_Rem: Remote detection of received security-critical data modified in transit

Identification by the remote site of the modification of security-critical (TSF) data occurring in transit from the remote site must occur.

O.TSF\_Rcv\_Err\_ID\_Rem is implemented in the TOE by:

1. FPT\_ITL.1: Inter-TSF detection of modification

Component Application: For the TOE: Modification metric: specify desired strength of function for received messages. (The modification metric for received TSF data must be compatible with that used by the remote site.)

Action if modification of imported data is detected: Specify notification of the remote site, and any other desired responses made by the TSF.

For the IT Environment: Modification metric: specify desired strength of function for exported data.

Action if modification of exported data is detected (and reported to the remote site by the remote site): Specify desired action.

O.TSF\_Snd\_Err\_ID\_Loc: Local detection of sent security-critical data modified in transit

Identification of modification of security-critical (TSF) data occurring in transit to a remote site by the TSF must occur.

O.TSF\_Snd\_Err\_ID\_Loc is implemented in the TOE by:

1. FPT\_ITL.1: Inter-TSF detection of modification

Component Application: For the TOE: Modification metric: specify desired strength of function for exported data.

Action if modification of exported data is detected (and reported to the TOE by the remote trusted product): Notify the remote site. This may be either automated or manual. In the latter case, there needs to be an administrative role for maintaining TSF data integrity.

For the IT Environment: Modification metric: specify desired strength of function for received messages. (The modification metric for received TSF data must be compatible with that used by the TOE.)

Action if modification of imported data is detected: Specify notification of the TOE, and any other desired responses made by the remote site.

Defined modification metric:

Action to be taken:

O.TSF\_Snd\_Err\_ID\_Rem: Remote detection of sent security-critical data modified in transit.

Identification of modification of security-critical (TSF) data occurring in transit to a remote site by the remote site must occur.

O.TSF\_Snd\_Err\_ID\_Rem is implemented in the TOE by:

1. FPT\_ITL.1: Inter-TSF detection of modification

Component Application: For the TOE: Modification metric: specify desired strength of function for exported data.

Action if modification of exported data is detected: not necessary.

For the IT Environment: Modification metric: specify desired strength of function for received messages. (The modification metric for received TSF data must be compatible with that used by the TOE.)

Action if modification of imported data is detected: Specify any desired responses made by the remote site.

Defined modification metric:

Action to be taken:

O.Tamper\_ID: Tamper detection

Provide system features that detect physical tampering of a system component, and use those features to limit security breaches.

Implementation Application: Two variations of this objective exist (choose one).

[Detect] This variation provides basic capabilities to detect physical tampering attacks. It relies more on user responsibility to follow procedures for identifying and acting upon detected physical tampering attacks.

[Notify] This variant increases TOE responsibility and decreases user responsibility, because the TSF alerts a designated individual when it has been tampered with.

O.Tamper\_ID is implemented in the TOE by:

1. AGD\_ADM.1: Administrator guidance

Component Application: Explain how to detect and report physical tampering or errors. At least for the [Detect] variant of Tamper\_ID, also explain when to look (e.g., periodically, when a user suspects tampering, when an unexpected error or specific type of breach has occurred, when a user may have violated responsibilities for the physical protection of the TOE).

2. AGD\_USR.1: User guidance

Component Application: Explain how to detect and report physical tampering or errors. At least for the [Detect] variant of Tamper\_ID, also explain when to look (e.g., periodically, when a user suspects tampering, when an unexpected error or specific type of breach has occurred, when a user may have violated responsibilities for the physical protection of the TOE).

3. FPT\_PHP.1: Passive detection of physical attack

Component Application: Environment: Some capability to determine when a physical tampering attack has occurred must be provided, but it is not necessary for the TOE to provide this functionality. The method provided should be supported by appropriate documentation.

Component Application Rationale: FPT\_PHP.1 provides basic physical tampering detection features.

Physical attacks may go undetected for longer periods of time.

O.Tamper\_Resistance: Tamper resistance

Prevent or resist physical tampering with specified system devices and components.

Implementation Application: [FPT\_PHP.3] Provides automatic response to physical attacks against resources deemed critical, thereby resisting those attacks.

O.Tamper\_Resistance is implemented in the TOE by:

1. FPT\_PHP.3: Resistance to physical attack

O.Trusted\_Path: Provide a trusted path

Provide a trusted path between the user and the system. Execution of a user-requested action must be made via a trusted path with the following properties:

- \* The path is logically distinct from, and cannot be confused with other communication paths (by either the user or the system).
- \* The path provides assured identification of its end points.

O.Trusted\_Path is implemented in the TOE by:

1. FTP\_ITC.1: Inter-TSF trusted channel
2. FTP\_TRP.1: Trusted path

O.Trusted\_Path is implemented by the following Non-IT measures :

The trusted path is the establishment of the ephemeral session from the TCPA specification. This session establishes that both endpoints are known (i.e. they both have knowledge of the authentication token) and each subsequent communication requires the proving of knowledge of the ephemeral session key.

O.Trusted\_Recovery: Trusted recovery of security functionality

Recovery to a secure state, without security compromise, after a discontinuity of operations.

Implementation Application: There are three variants of this objective (choose one):

[Manual]: Choose FPT\_RCV.1

[Automated]: Choose FPT\_RCV.2

[Quantified]: Choose FPT\_RCV.3

The [Manual] variant should be chosen when manual procedures for recovery are acceptable.

The [Automated] variant should be chosen when the TOE must recover from some failure scenarios without human intervention.

The [Quantified] variant should be chosen following the same criteria as for the Automated variant, but in addition there is a strong need to limit the loss of TSF data during failure scenarios to strictly defined limits, and/or to be able to determine exactly



what TSF data could not be restored.

---

For the [Manual] variant, failure scenarios should have a relatively infrequent occurrence and longer "down times" must be acceptable.

The [Automated] variant is justified when: (a) there is a higher risk from failure scenarios, (b) it is less acceptable for the TOE to occasionally transit from an operational mode to a manual mode, and (c) it is less feasible for human operators to intervene.

O.Trusted\_Recovery is implemented in the TOE by:

1. FPT\_RCV.3: Automated recovery without undue loss

Component Application: To support the implementation of this requirement, the developer must provide a definition of "secure state" so that the requirement can be evaluated. This definition can be provided through the associated dependency on ADV\_SPM.1.

Component Application Rationale: The definition of a "secure state" should be provided by the security model documentation (ADV\_SPM.1) when a model is required by the PP. If the requirement for a security model is not present, it is still necessary for the developer to provide this definition in some manner. How this definition is provided should be identified in the rationale that defends the exclusion of ADV\_SPM.1.

The degree of this self-protection is directly proportional to the scope of failures for which the TOE provides automatic recovery in relation to the scope of all possible errors.

This component provides additional protection by limiting the loss of TSF data (to a quantified limit) and providing a capability to determine which TSF data cannot be recovered.

O.Trusted\_Recovery\_Doc: Documentation of untrusted data recovery  
Provide trusted recovery to ensure that data cannot be lost or misplaced. Any circumstances which can cause untrusted recovery to be documented with mitigating procedures established.

O.Trusted\_Recovery\_Doc is implemented in the TOE by:

1. AGD\_ADM.1: Administrator guidance

Component Application: Provide administrative guidance to allow the administrators to determine when recovery fails.

O.User\_Auth\_Management: User authorization management  
Manage and update user authorization and privilege data in accordance with organizational security and personnel policies.

O.User\_Auth\_Management is implemented in the TOE by:

1. AGD\_ADM.1: Administrator guidance

Component Application: The administrator must be provided guidance on how to properly adjust a user's authorization status.

Component Application Rationale: This administrative guidance is needed to support secure user authorization management.

2. AGD\_USR.1: User guidance

3. FMT\_MSA.1: Management of security attributes

Component Application: All user security attributes that requires control should be specified, along with permitted operations and those roles that the system will allow to perform those operations.

Component Application Rationale: This component provides the basic capability to update user security attributes. Additional procedural controls may be required to supplement the controls provided by this component. For instance, there should be a procedural control to ensure administrators learn of necessary changes a user's authorization status.

4. FMT\_REV.1: Revocation

Component Application: This component provides users and or administrators with the capability to revoke attributes for users (or possibly the resources they may access), as specified.

Component Application Rationale: This component provides the basic capability for an administrator to revoke security attributes to objects controlled by that user.

O.User\_Conf\_Prevention: Basic confidentiality-breach prevention  
Prevent unauthorized export of confidential information from the TOE with moderate effectiveness.

Implementation Application: This objective is implemented in terms of a technical security policy, which is referred to as the observation-control policy. This observation-control policy may be cast as either an access control or information-flow policy. When cast as an access control policy, use the components from FDP\_ACC and FDP\_ACF.

When cast as an information flow policy, use the components from FDP\_IFC and FDP\_IFF. In both cases, use of FDP\_ETC is required to meet this objective.

O.User\_Conf\_Prevention is implemented in the TOE by:

1. FDP\_ACF.1: Security attribute based access control
2. FDP\_IFF.1: Simple security attributes

Component Application: Information flow control SFP: the above observation-control policy

The minimum number and type of security attributes: readership attributes denoting a set of allowed readers or a set of allowed roles

For each operation, the security attribute-based relationship that must hold:

Additional information flow control SFP rules: a subject may observe or receive information only if it is acting on behalf of a user possessing the information's readership attributes; a subject may transmit or send information only if the readership of the information contains the readership of all the information that the subject has acquired

Component Application Rationale: These security attributes may take the form of sensitivity labels drawn from a partially ordered set of security levels.

The attributes apply to both subjects and information. In traditional implementations, information attributes are not derived from the information itself, but reside with the information in storage objects.

3. FDP\_ACC.2: Complete access control

O.User\_Data\_Integrity: Integrity protection of stored user data  
Provide appropriate integrity protection for stored user data.

O.User\_Data\_Integrity is implemented in the TOE by:

1. FDP\_SDI.2: Stored data integrity monitoring and action

O.User\_Data\_Transfer: Protection of transmitted user data  
Provide the ability to have physically protected communications lines, intrusion detection for communications lines, and/or need-to-know isolation for communications lines.

O.User\_Data\_Transfer is implemented in the TOE by:

1. FDP\_ITT.2: Transmission separation by attribute

O.User\_Defined\_AC: User-defined access control

Enforce an access control policy whereby users may determine who may access information they control.

Implementation Application: Two variants of this objective can be implemented:  
[Subset] Choose FDP\_ACC.1 and FDP\_ACF.1. This variant allows subsets within the system to be constrained by the defined controls while others need not be. Subsets are defined by specific data types, users, and/or operations, or combinations thereof. This variant provides flexibility at the expense of more comprehensive and consistent protection.

[Complete] Choose FDP\_ACC.2 and FDP\_ACF.1. This variant provides access control between all users, data types, and operations on the system, although the exact controls that must be applied need not be identical. For example, the controls placed on "public" files need not be equivalent with those placed on "project" files. This variant attempts to provide comprehensive and consistent protection but generally requires much more rigor to implement.

O.User\_Defined\_AC is implemented in the TOE by:

1. FDP\_ACC.2: Complete access control

Component Application: Access control SFP: Provide an identifier for an access control SFP for to which the scope of policy enforcement (defined in terms of subjects and objects, below) applies. Other component specified throughout the PP will use this identifier as a means to associate applicability. In particular, the dependent FDP\_ACF component will define the applicable policy enforcement by referencing this identifier.

List of subjects and objects: List the specific subjects and objects that are within the scope of this policy.

Component Application Rationale: This component identifies and defines a scope for the identified policy. The policy identifier is used to associate related TSFs (e.g., that are constrained by the same scope and rules). The rationale should explain the relevance of listed subjects, and objects in terms of Security Environment abstractions. All operations between the listed subjects and objects are subject to the defined SFP.

The rationale should explain how the specifications within the completed operations serve to address the goals of the associated Objective, as well as any specific coverage of Security Environment concerns that they may provide.

2. FDP\_ACF.1: Security attribute based access control

Component Application: Access control SFP: Name the associated access control policy (as identified within the dependent FDP\_ACC component) that are applicable for the attributes, rules, etc. specified below.

Security attributes: List the attributes here that must be used to support enforcement of the rules listed below.

Rules governing access: List the rules that express the intent and scope of that policy component.

Rules that explicitly authorize access: List the rules that demonstrate conditions (e.g., specific security attribute values) that result in an explicit "grant" authorization.

Be sure to avoid and/or clarify any contradictions or ambiguities that may result between an explicit "grant" rule and an explicit "deny" rule.

Component Application Rationale: The policy enforcement specified in this component serves to associate the completed operations that list attributes, rules, etc. with the policy identified in the dependent FDP\_ACC component. These completed operations are essentially the defined enforcement requirements for the associated policy.

The nature of the rules that must be specified will vary depending on which policy (FDP\_ACC) component is chosen.

This operation allows a convenient expression of "grant" exceptions to the general rules above. Care should be taken that the intent and scope of the associated policy are not violated with automatic grant exceptions. However, this operation allows the efficient expression of some rules that are generally considered safe (e.g., always granting access to "owner" subjects).

The explicit "denial" rules allow specification of the exceptions to the general rules above. Unlike exceptional "grant" rules, these types of exceptions are generally safe in that the scope of typical policy enforcement cannot be circumvented through denials of access. This may not be the case when Availability concerns are taken into account.

Contradictions and/or ambiguities in the rules specification may result in ineffective policy enforcement.

The rationale should explain how the specifications within the completed operations serve to address the goals of the associated Objective, as well as any specific coverage of Security Environment concerns that they may provide.

O.User\_Guidance: User guidance documentation  
Provide documentation for the general user.

O.User\_Guidance is implemented in the TOE by:

1. AGD\_USR.1: User guidance

Component Application: Associated information must appear in User Guidance documentation.

Component Application Rationale: The rationale should address how the specific information that is to be provided in user guidance documentation will address the Security Environment concerns to which the O.User\_Guidance objective is mapped.

Security Objectives:

Security Objectives is implemented in the TOE by:

1. ATE\_COV.2: Analysis of coverage

### 6.3.2 - Assurance Security Requirements Rationale

Provide rationale for chosen assurance level.

## 6.4 - Dependency Rationale

**Table 6-4 Functional and Assurance Requirements Dependencies**

Requirement	Dependencies
Functional Requirements	
FAU_ARP.1	FAU_SAA.1
FAU_GEN.1	Rationale provided for: ( FPT_STM.1 )
FAU_SAA.1	FAU_GEN.1
FAU_SAR.1	FAU_GEN.1
FAU_SEL.1	FAU_GEN.1, FMT_MTD.1

FAU_STG.1	FAU_GEN.1
FCO_NRO.1	FIA_UID.1
FCO_NRO.2	FIA_UID.1
FCO_NRR.1	FIA_UID.1
FCO_NRR.2	FIA_UID.1
FCS_CKM.1	FCS_COP.1, FCS_CKM.4, FMT_MSA.2
FCS_CKM.4	FDP_ITC.1, FCS_CKM.1, FMT_MSA.2
FCS_COP.1	FDP_ITC.1, FCS_CKM.1, FCS_CKM.4, FMT_MSA.2
FDP_ACC.2	FDP_ACF.1
FDP_ACF.1	FDP_ACC.1, FMT_MSA.3
FDP_ETC.2	FDP_ACC.1, FDP_IFC.1
FDP_IFC.2	FDP_IFF.1
FDP_IFF.1	FDP_IFC.1, FMT_MSA.3
FDP_ITC.1	FDP_ACC.1, FDP_IFC.1, FMT_MSA.3
FDP_ITC.2	FDP_ACC.1, FDP_IFC.1, FTP_ITC.1, FTP_TRP.1, FPT_TDC.1
FDP_ITT.2	FDP_ACC.1, FDP_IFC.1
FDP_ITT.4	FDP_ACC.1, FDP_IFC.1, FDP_ITT.2
FDP_ROL.1	FDP_ACC.1, FDP_IFC.1
FDP_UCT.1	FTP_ITC.1, FTP_TRP.1, FDP_ACC.1, FDP_IFC.1
FDP_UIT.1	FDP_ACC.1, FDP_IFC.1, FTP_ITC.1, FTP_TRP.1
FIA_AFL.1	FIA_UAU.1
FIA_UAU.2	FIA_UID.1
FMT_MOF.1	FMT_SMR.1
FMT_MSA.1	FDP_ACC.1, FDP_IFC.1, FMT_SMR.1

FMT_MSA.2	ADV_SPM.1, FDP_ACC.1, FDP_IFC.1, FMT_MSA.1, FMT_SMR.1
FMT_MSA.3	FMT_MSA.1, FMT_SMR.1
FMT_MTD.1	FMT_SMR.1
FMT_MTD.2	FMT_MTD.1, FMT_SMR.1
FMT_MTD.3	ADV_SPM.1, FMT_MTD.1
FMT_REV.1	FMT_SMR.1
FPT_FLS.1	ADV_SPM.1
FPT_PHP.1	FMT_MOF.1
FPT_RCV.3	FPT_TST.1, AGD_ADM.1, ADV_SPM.1
FPT_RCV.4	ADV_SPM.1
FPT_SSP.2	FPT_ITT.1
FPT_TRC.1	FPT_ITT.1
FPT_TST.1	FPT_AMT.1
FRU_FLT.1	FPT_FLS.1
FTA_MCS.1	FIA_UID.1
Assurance Requirements	
ACM_CAP.3	ACM_SCP.1, ALC_DVS.1
ACM_SCP.1	ACM_CAP.3
ADO_IGS.1	AGD_ADM.1
ADV_FSP.1	ADV_RCR.1
ADV_HLD.2	ADV_FSP.1, ADV_RCR.1
ADV_SPM.1	ADV_FSP.1
AGD_ADM.1	ADV_FSP.1
AGD_USR.1	ADV_FSP.1



ATE_COV.2	ADV_FSP.1, ATE_FUN.1
ATE_DPT.1	ADV_HLD.1, ATE_FUN.1
ATE_IND.2	ADV_FSP.1, AGD_ADM.1, AGD_USR.1, ATE_FUN.1
AVA_MSU.1	ADO_IGS.1, ADV_FSP.1, AGD_ADM.1, AGD_USR.1
AVA_SOF.1	ADV_FSP.1, ADV_HLD.1
AVA_VLA.1	ADV_FSP.1, ADV_HLD.1, AGD_ADM.1, AGD_USR.1

### Justification of Unsupported Dependencies

#### Dependency of FAU\_GEN.1 on FPT\_STM.1

The TPM merely generates the audit event it does add the event into the log. The TPS must collect the audit event from the TPM and associate the event with a time stamp. The TPM has no source of time according to the 1.0 version of the specification.

## 6.5 - Security Functional Requirements Grounding in Objectives

**Table 6-5 Requirements to Objectives Mapping**

Requirements	Objectives
ACM_CAP.3	O.SpecRef
ACM_SCP.1	O.SpecRef
ADO_DEL.1	O.Apply_Code_Fixes, O.SpecRef
ADO_IGS.1	O.SpecRef
ADV_FSP.1	O.Crypto_Key_Man, O.Crypto_Modular_Dsgn, O.Crypto_Operation, O.SpecRef, O.NoBore, O.Protected_Capability, O.RootMeasurement, O.MetricReporting, O.RootReporting, O.EMSEC_Design, O.IntelEman_Control, O.IntelEman_Contain, O.Standard_Output_Pres
ADV_HLD.2	O.Crypto_Dsgn_Impl, O.SpecRef, O.NoBore, O.Protected_Capability, O.RootMeasurement

	O.MetricReporting, O.RootReporting, O.EMSEC_Design, O.IntelEman_Control, O.IntelEman_Contain, O.Standard_Output_Pres
ADV_RCR.1	O.Crypto_Dsgn_Impl, O.Source_Code_Exam, O.SpecRef
ADV_SPM.1	O.Crypto_Key_Man, O.Crypto_Operation, O.SpecRef, O.RootReporting, O.NoBore
AGD_ADM.1	O.Admin_Guidance, O.Apply_Code_Fixes, O.Crypto_Import_Export, O.Hack_Limit_Sessions, O.NonRepud_Assess_Recd, O.NonRepud_Assess_Sent, O.NonRepud_Gen_Recd, O.NonRepud_Gen_Sent, O.React_Discovered_Atk, O.Tamper_ID, O.Trusted_Recovery_Doc, O.User_Auth_Management, O.SpecRef
AGD_USR.1	O.Crypto_Import_Export, O.Manage_Res_Sec_Attr, O.NonRepud_Assess_Recd, O.NonRepud_Assess_Sent, O.Tamper_ID, O.User_Auth_Management, O.User_Guidance, O.SpecRef
ALC_DVS.1	O.Lifecycle_Security, O.SpecRef, O.SecureManufacturing
ALC_LCD.1	O.Lifecycle_Security, O.SpecRef
ATE_COV.2	O.SpecRef, Security Objectives, O.EMSEC_Design
ATE_DPT.1	O.Crypto_Test_Reqs, O.SpecRef, O.EMSEC_Design
ATE_FUN.1	O.Crypto_Test_Reqs, O.Sys_Assur_HW/SW/FW, O.SpecRef, O.EMSEC_Design
ATE_IND.2	O.SpecRef, O.EMSEC_Design, O.IntelEman_Contain, O.IntelEman_Control
AVA_MSU.1	O.SpecRef
AVA_SOF.1	O.SpecRef
AVA_VLA.1	O.Crypto_Key_Man, O.Crypto_Test_Reqs, O.SpecRef
FAU_ARP.1	O.React_Discovered_Atk
FAU_GEN.1	O.Audit_Generation, O.Manage_Res_Sec_Attr, O.AuditLog

FAU_SAA.1	O.AuditLog
FAU_SAR.1	O.AuditLog
FAU_SEL.1	O.AuditLog
FAU_STG.1	O.Audit_Protect
FCO_NRO.1	O.NonRepud_Assess_Sent, O.NonRepud_Gen_Sent, O.MessageAuthentication
FCO_NRO.2	O.NonRepud_Gen_Sent, O.MessageAuthentication
FCO_NRR.1	O.NonRepud_Assess_Recd, O.NonRepud_Gen_Recd, O.MessageAuthentication
FCO_NRR.2	O.NonRepud_Gen_Recd, O.MessageAuthentication
FCS_CKM.1	O.Crypto_Key_Man, O.SpecRef, O.TCPAIdentities, O.Protected_Capability
FCS_CKM.4	O.Crypto_Key_Man, O.SpecRef, O.TCPAIdentities, O.Protected_Capability
FCS_COP.1	O.Crypto_Operation, O.Data_Exchange_Conf, O.MessageAuthentication, O.Protected_Capability
FDP_ACC.2	O.User_Defined_AC, O.MessageAuthentication, O.TCPAProtectedStorage, O.AC_Label_Export, O.Admin_Code_Val, O.Crypto_AC, O.Crypto_Key_Man, O.Export_Control, O.Input_Inspection, O.Obj_Attr_Integrity, O.Obj_Protection, O.User_Conf_Prevention
FDP_ACF.1	O.AC_Label_Export, O.Admin_Code_Val, O.Crypto_AC, O.Crypto_Key_Man, O.Input_Inspection, O.Obj_Attr_Integrity, O.Obj_Protection, O.User_Conf_Prevention, O.User_Defined_AC, O.MessageAuthentication, O.TCPAProtectedStorage
FDP_DAU.1	O.Change_Control_Users, O.MessageAuthentication, O.TCPAProtectedStorage
FDP_ETC.2	O.AC_Label_Export, O.Clean_Obj_Recovery, O.Code_Signing, O.Crypto_Import_Export, O.Data_Export_Control, O.Data_Imp_Exp_Control, O.External_Labels, O.Integ_Data_Mark_Exp

	O.TCPAProtectedStorage
FDP_IFC.2	O.Info_Flow_Control, O.TCPAProtectedStorage
FDP_IFF.1	O.Data_Imp_Exp_Control, O.Info_Flow_Control, O.User_Conf_Prevention, O.TCPAProtectedStorage
FDP_ITC.1	O.Clean_Obj_Recovery, O.Crypto_Import_Export, O.Crypto_Key_Man, O.Input_Inspection, O.Malicious_Code, O.TCPAProtectedStorage
FDP_ITC.2	O.Code_Signing, O.Crypto_Import_Export, O.Data_Imp_Exp_Control, O.External_Labels, O.TCPAProtectedStorage
FDP_ITT.2	O.Integ_User_Data_Int, O.TCPAProtectedStorage, O.User_Data_Transfer
FDP_ITT.4	O.SpecRef, O.Protected_Capability
FDP_RIP.1	O.No_Residual_Info, O.TCPAProtectedStorage
FDP_RIP.2	O.No_Residual_Info, O.TCPAProtectedStorage
FDP_ROL.1	O.Clean_Obj_Recovery, O.TCPAProtectedStorage
FDP_SDI.2	O.Integrity_Data/SW, O.TCPAProtectedStorage, O.Admin_Code_Val, O.Crypto_Self_Test, O.General_Integ_Checks, O.Storage_Integrity, O.User_Data_Integrity
FDP_UCT.1	O.TCPAProtectedStorage
FDP_UIT.1	O.Code_Signing, O.Rcv_MsgMod_ID, O.Snt_MsgMod_ID, O.TCPAProtectedStorage
FIA_AFL.1	O.MessageAuthentication, O.TCPAProtectedStorage
FIA_UAU.2	O.Limit_Actions_Auth, O.MessageAuthentication, O.TCPAProtectedStorage
FIA_UAU.4	O.MessageAuthentication
FIA_UID.2	O.MessageAuthentication
FMT_MOF.1	O.Apply_Code_Fixes, O.Config_Management, O.NonRepud_Access_Recd O.NonRepud_Access_Sent

	O.NonRepud_Gen_Recd, O.NonRepud_Gen_Sent, O.Security_Func_Mgt, O.Clean_Obj_Recovery
FMT_MSA.1	O.Admin_Code_Val, O.Apply_Code_Fixes, O.Crypto_Key_Man, O.Hack_Limit_Sessions, O.Manage_Res_Sec_Attr, O.Obj_Attr_Integrity, O.Security_Attr_Mgt, O.User_Auth_Management
FMT_MSA.2	O.Obj_Attr_Integrity, O.Security_Attr_Mgt
FMT_MSA.3	O.Obj_Attr_Integrity, O.Obj_Protection, O.Security_Attr_Mgt
FMT_MTD.1	O.Config_Management, O.Crypto_Key_Man, O.Security_Data_Mgt
FMT_MTD.2	O.Manage_TSF_Data, O.Security_Data_Mgt
FMT_MTD.3	O.Security_Data_Mgt
FMT_REV.1	O.User_Auth_Management
FMT_SMR.2	O.Security_Roles, O.Obj_Attr_Integrity
FPR_ANO.2	O.TCPAProtectedStorage
FPR_UNL.1	O.Prevent_Link
FPT_AMT.1	O.Crypto_Data_Sep, O.Crypto_Self_Test, O.Crypto_Test_Reqs, O.Integrity_Practice, O.Malicious_Code, O.Sys_Backup_Verify
FPT_FLS.1	O.Crypto_Self_Test, O.Fail_Secure, O.Secure_State
FPT_ITI.1	O.Integ_Sys_Data_Ext, O.TSF_Rcv_Err_ID_Loc, O.TSF_Rcv_Err_ID_Rem, O.TSF_Snd_Err_ID_Loc, O.TSF_Snd_Err_ID_Rem
FPT_ITT.2	O.SpecRef, O.Integ_Sys_Data_Int
FPT_PHP.1	O.Malicious_Code, O.Sys_Backup_Verify, O.Tamper_ID, O.SpecRef
FPT_PHP.3	O.Tamper_Resistance, O.SpecRef
FPT_PHP_EMSEC_Design	O.EMSEC_Design

FPT_RCV.3	O.Secure_State, O.Sys_Backup_Procs, O.Trusted_Recovery
FPT_RCV.4	O.Atomic_Functions, O.Secure_State
FPT_RPL.1	O.MessageAuthentication, O.SpecRef
FPT_SEP.2	O.Crypto_Data_Sep, O.Maintain_Sec_Domain, O.Sys_Self_Protection, O.Crypto_Key_Man
FPT_SSP.2	O.Integ_Sys_Data_Int
FPT_TDC.1	O.SpecRef
FPT_TRC.1	O.Integrity_Data_Rep
FPT_TST.1	O.Admin_Code_Val, O.Crypto_Self_Test, O.General_Integ_Checks, O.Integrity_Practice, O.Malicious_Code, O.Sys_Assur_HW/SW/FW, O.Sys_Backup_Verify, O.Clean_Obj_Recovery
FRU_FLT.1	O.Fault_Tolerance
FRU_PRS.1	O.SpecRef
FTA_MCS.1	O.Limit_Comm_Sessions
FTP_ITC.1	O.Trusted_Path, O.Crypto_Import_Export
FTP_TRP.1	O.Trusted_Path

## 6.6 - Rationale for Extensions

### Rationale for Extension FPT\_PHP\_EMSEC\_Design to FPT\_PHP

IT equipment that processes sensitive user data may need to avoid the unintended transmission of information-bearing sounds or electromagnetic signals.

## Appendix A - Acronyms

CC - Common Criteria  
EAL - Evaluation Assurance Level

PP - Protection Profile  
SF - Security Function  
SFP - Security Function Policy  
SOF - Strength of Function  
ST - Security Target  
TOE - Target of Evaluation  
TSC - TSF Scope of Control  
TSF - TOE Security Functions  
TSFI - TSF Interface  
TSP - TOE Security Policy