CONTROLLED ACCESS PROTECTION PROFILE

Version 1.d

Information Systems Security Organization

National Security Agency
9800 Savage Road
Fort George G. Meade, MD 20755-6000

8 October 1999

Foreword

This publication, “Controlled Access Protection Profile”, is issued by the Information Systems Security Organization (ISSO) as part of its program to promulgate security standards for information systems. This protection profile is based on the C2 class of the “Department of Defense Trusted Computer System Evaluation Criteria” (DOD 5200.28-STD), and supersede that class for use in evaluations.

The base set of requirements used in this protection profile are taken from the “Common Criteria for Information Technology Security Evaluations, Version 2.0.” Further information, including the status and updates, of both this protection profile and the Common Criteria can be found on the internet at “http://www.radium.ncsc.mil/tpep”.

Comments on this document should be directed to:

Protection Profile Registry (V13)
National Security Agency
9800 Savage Road, Suite 6740
Ft. George G. Meade, MD 20755-6740

or

PPRegistry@gibraltar.ncsc.mil

or

(410) 854-4458

_____

Keith Brewster 8 October 1999
Chief
Information Assurance Criteria Support

Revision History

CAPP 1.0 - Based on version 1.0 of the Common Criteria. Also known as the C2 Protection Profile.

CAPP 1.a - Completely revised version based on version 2.0 of the Common Criteria.

CAPP 1.b - Major updates to the policy and objective sections. Also both pre and post selection of audit events were included.

LSPP 1.0 - Baseline version, derived from CAPP 1.b.

CAPP 1.c - Changes based on comments from external review of the document. Changes include:

- Reformatting the list of audit events into a table;
- FAU_STG.2 was changed to FAU_STG.1;
- Security-relevant authorizations were moved to roles section.
- The O.ENFORCEMENT objective was clarified.

LSPP 1.a - Updates include all those made for CAPP 1.c, plus the following:

- Management of MAC related attributes added.

Table of Contents

1.0 Introduction

This section contains document management and overview information necessary to allow a Protection Profile (PP) to be registered through a Protection Profile Registry. The PP *identification* provides the labelling and descriptive information necessary to identify, catalogue, register, and cross-reference a PP. The PP *overview* summarizes the profile in narrative form and provides sufficient information for a potential user to determine whether the PP is of interest. The overview can also be used as a stand-alone abstract for PP catalogues and registers. The conventions section provides an explanation of how this document is organized and the terms section gives a basic definition of terms which are specific to this PP.

1.1 Identification

Title: Controlled Access Protection Profile (CAPP)

Registration: Information Systems Security Organization (ISSO)

Keywords: access control, discretionary access control, general-purpose operating system, information protection

1.2 Overview

The Common Criteria (CC) Controlled Access Protection Profile, hereafter called CAPP, specifies a set of security functional and assurance requirements for Information Technology (IT) products. CAPP-conformant products support access controls that are capable of enforcing access limitations on individual users and data objects. CAPP-conformant products also provide an audit capability which records the security-relevant events which occur within the system.

The CAPP provides for a level of protection which is appropriate for an assumed non-hostile and well-managed user community requiring protection against threats of inadvertent or casual attempts to breach the system security. The profile is not intended to be applicable to circumstances in which protection is required against determined attempts by hostile and well funded attackers to breach system security. The CAPP does not fully address the threats posed by malicious system development or administrative personnel. CAPP-conformant products are suitable for use in both commercial and government environments.

The CAPP was derived from the requirements of the C2 class of the U.S. Department of Defense (DoD) *Trusted Computer System Evaluation Criteria (TCSEC)*, dated December, 1985, and the material upon which those requirements are based. This protection profile provides security functions and assurances which are equivalent to those provided by the TCSEC and replaces the requirements used for C2 trusted product evaluations.

The CAPP is generally applicable to distributed systems but does not address the security requirements which arise specifically out of the need to distribute the resources within a network.

1.3 Strength of Environment

The CAPP is for a generalized environment with a moderate level of risk to the assets. The assurance requirements and the minimum strength of function were chosen to be consistent with that level of risk. The assurance level is EAL 3 and the minimum strength of function is SOF-medium.

1.4 Conventions

This document is organized based on Annex B of Part 1 of the Common Criteria. There are several deviations in the organization of this profile. First, rather than being a separate section, the application notes have been integrated with requirements and indicated as notes. Likewise, the rationale has been

integrated where appropriate.

For each component, an application note may appear. Application notes document guidance for how the requirement is expected to be applied. For additional guidance, the CC itself should be consulted. Following the application note is rationale for the inclusion of the component in the requirement set.

In the requirement sections, each section which represents a requirement family or component, there is a mnemonic in parenthesis. These refer to the requirement section in the CC from which it was derived. Requirement elements have these references includes as superscripted text at the end of the element. In some places these references indicate a note instead. These notes represent components or elements which do not appear in the CC, and an explanation can be found in section 8.0 of this profile. The superscripted text which appears in the audit event list in 5.1.1.1 is cross referenced to the functional requirement component in this profile upon which that event was derived.

1.5 Terms

This profile uses the following terms which are described in this section to aid in the application of the requirements:

- User
- Authorized User
- Authorized Administrator
- Discretionary Access Control (DAC) Policy
- Mediation
- Access
- Authorization

A *user* is an individual who attempts to invoke a service offered by the TOE.

An *authorized user* is a user who has been properly identified and authenticated. These users are considered to be legitimate users of the TOE.

An *authorized administrator* is an authorized user who has been granted the authority to manage the TOE. These users are expected to use this authority only in the manner prescribed by the guidance given them.

The Discretionary Access Control Policy, also referred to as DAC, is the basic policy that a CAPP conformant TOE enforces over users and resources.

Whether a user is granted a requested action is determined by the TOE Security Policy (TSP) which is specified in this profile in the context of Discretionary Access Control (DAC). The *DAC policy* is the set of rules used to mediate user access to TOE protected objects and can be generally characterized as a policy which requires the TOE to allow authorized users and authorized administrators to control access to objects based on individual user identification. When the DAC policy rules are invoked, the TOE is said to be *mediating* access to TOE protected objects. However, there may be instances when the DAC policy is not invoked meaning that there may be objects residing in the TOE which are not protected by the TSP. In these instances the TOE is said to not be mediating access to a set of objects even though the TOE is executing a (possibly unauthorized) user request.

The DAC policy consists of two types of rules: those which apply to the behavior of authorized users (termed access rules) and those which apply to the behavior of authorized administrators (termed authorization rules). If an authorized user is granted a request to operate on an object, the user is said to have *access* to that object. There are numerous types of access; typical ones include read access and write access which allow the reading and writing of objects respectively. If an authorized administrator is granted a requested service, the user is said to have *authorization* to the requested service or object. As for access, there are numerous possible authorizations. Typical authorizations include auditor authorization which allows an administrator to view audit records and execute audit tools and DAC override authorization which allows an administrator to override object access controls

to administer the system.

2.0 TOE Description

The CAPP defines a set of security requirements to be levied on Targets of Evaluation (TOEs). These TOEs include information systems which contain general-purpose operating systems, such as workstations, mainframes, or personal computers. These systems can be comprised of a single host or a set of cooperating hosts in a distributed system. Such systems permit one or more processors along with peripherals and storage devices to be used by multiple users to perform a variety of functions requiring controlled, shared access to the information stored on the system. Such installations are typical of personal, work group, or enterprise computing systems accessed by users local to, or with otherwise protected access to, the computer systems.

The CAPP is applicable to TOEs that provide facilities for on-line interaction with users, as well as TOEs that provide for batch processing. The protection profile is also generally applicable to TOEs incorporating network functions but contains no network specific requirements. Networking is covered only to the extent to which the TOE can be considered to be part of a centrally-managed system that meets a common set of security requirements.

The CAPP assumes that responsibility for the safeguarding of the data protected by the TOEs security functions (TSF) can be delegated to the TOE users. All data is under the control of the TOE. The data are stored in objects, and the TSF can associate with each controlled object a description of the access rights to that object.

All individual users are assigned a unique identifier. This identifier supports individual accountability. The TSF authenticates the claimed identity of the user before allowing the user to perform any actions that require TSF mediation, other than actions which aid an authorized user in gaining access to the TOE.

3.0 Security Environment

3.1 Threats

The CAPP has derived all security objectives from the statement of Organizational Security Policy found in the following section. Therefore, there is no statement of the explicit threats countered by the CAPP.

3.2 Organizational Security Policies

An Organizational Security Policy is a set of rules or procedures imposed by an organization upon its operations to protect its sensitive data. Although the organizational security policies described below is drawn from DoD Manual 5200.28-M (Techniques and procedures for Implementing, Deactivating and Evaluating Resource Sharing ADP Systems) it applies to many non-DoD environments.

P.AUTHORIZED_USERS

Only those users who have been authorized to access the information within the system may access the system.

P.NEED_TO_KNOW

The system must limit the access to, modification of, and destruction of the information in protected resources to those authorized users which have a “need to know” for that information.

P.ACCOUNTABILITY

The users of the system shall be held accountable for their actions within the system.

3.3 Security Usage Assumptions

This section describes the security aspects of the environment in which the TOE will be, or is intended to be used. This includes information about the physical, personnel, and connectivity aspects of the environment.

A CAPP-conformant TOE is assured to provide effective security measures in a cooperative non-hostile environment only if it is installed, managed, and used correctly. The operational environment must be managed in accordance with assurance requirements documentation for delivery, operation, and user/administrator guidance. The following specific conditions are assumed to exist in an environment where CAPP-conformant TOEs are employed.

3.3.1 Physical Assumptions

CAPP-conformant TOEs are intended for application in user areas that have physical control and monitoring. It is assumed that the following physical conditions will exist:

A.LOCATE

The processing resources of the TOE will be located within controlled access facilities which will prevent unauthorized physical access.

A.PROTECT

The TOE hardware and software critical to security policy enforcement will be protected from unauthorized physical modification.

3.3.2 Personnel Assumptions

It is assumed that the following personnel conditions will exist:

A.MANAGE

There will be one or more competent individuals assigned to manage the TOE and the security of the information it contains.

A.NO_EVIL_ADM

The system administrative personnel are not careless, willfully negligent, or hostile, and will follow and abide by the instructions provided by the administrator documentation.

A.COOP

Authorized users possess the necessary authorization to access at least some of the information managed by the TOE and are expected to act in a cooperating manner in a benign environment.

3.3.3 Connectivity Assumptions

The CAPP contains no explicit network or distributed system requirements. However, it is assumed that the following connectivity conditions exist:

A.PEER

Any other systems with which the TOE communicates are assumed to be under the same management control and operate under the same security policy constraints. CAPP-conformant TOEs are applicable to networked or distributed environments only if the entire network operates under the same constraints and resides within a single management domain. There are no security requirements which address the need to trust external systems or the communications links to such systems.

A.CONNECT

All connections to peripheral devices reside within the controlled access facilities. CAPP-conformant TOEs only address security concerns related to the manipulation of the TOE through its authorized access points. Internal communication paths to access points such as terminals are assumed to be adequately protected.

4.0 Security Objectives

This section defines the security objectives of the TSF and its supporting environment. Security objectives, categorized as either IT security objectives or non-IT security objectives, reflect the stated intent to counter identified threats and/or comply with any organizational security policies identified. All of the identified threats and organizational policies are addressed under one of the categories below.

4.1 IT Security Objectives

The following are the CAPP IT security objectives:

O.AUTHORIZATION

The TSF must ensure that only authorized users gain access to the TOE and its resources.

O.DISCRETIONARY_ACCESS

The TSF must control accessed to resources based on identity of users. The TSF must allow authorized users to specify which resources may be accessed by which users.

O.AUDITING

The TSF must record the security relevant actions of users of the TOE. The TSF must present this information to authorized administrators.

O.RESIDUAL_INFORMATION

The TSF must ensure that any information contained in a protected resource is not released when the resource is recycled.

O.MANAGE

The TSF must provide all the functions and facilities necessary to support the authorized administrators that are responsible for the management of TOE security.

O.ENFORCEMENT

The TSF must be designed and implemented in a manner which ensures that the organizational policies are enforced in the target environment.

4.2 Non-IT Security Objectives

A CAPP-conformant TOE is assumed to be complete and self-contained and, as such, is not dependent upon any other products to perform properly. However, certain objectives with respect to the general operating environment must be met. The following are the CAPP non-IT security objectives:

O.INSTALL

Those responsible for the TOE must ensure that the TOE is delivered, installed, managed, and operated in a manner which maintains IT security objectives.

O.PHYSICAL

Those responsible for the TOE must ensure that those parts of the TOE critical to security policy are protected from physical attack which might compromise IT security objectives.

O.CREDEN

Those responsible for the TOE must ensure that all access credentials, such as passwords or other authentication information, are protected by the users in a manner which maintains IT security objec-

tives.

5.0 Functional Requirements

This chapter defines the functional requirements for the TOE. Functional requirements components in this profile were drawn from Part 2 of the CC. Some functional requirements are extensions to those found in the CC.

CC defined operations for assignment, selection, and refinement were used to tailor the requirements to the level of detail necessary to meet the stated security objectives. These operations are indicated through the use of underlined (assignments and selections) and italicized (refinements) text. All required operations not performed within this profile are clearly identified and described such that they can be correctly performed upon instantiation of the PP into a Security Target (ST) specification.

5.1 Security Audit (FAU)

5.1.1 Audit Data Generation (FAU_GEN.1)

5.1.1.1 The TSF shall be able to generate an audit record of the auditable events *listed in column "Event" of Table 1 (Auditable Events). This includes all auditable events for the basic level of audit, except FIA_UID.1's user identity during failures.* FAU_GEN.1.1 / NOTE 4

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

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

b) The additional information specified in the "Details" column of Table 1 (Auditable Events).

Application Note: For some situations it is possible that some events cannot be automatically generated. This is usually due to the audit functions not being operational at the time these events occur. Such events need to be documented in the Administrative Guidance, along with recommendation on how manual auditing should be established to cover these events.

Rationale: This component supports O.AUDITING by specifying the detailed, security-relevant events and data that the audit mechanism must be capable of generating and recording. The "basic" level of auditing was selected as best representing the "mainstream" of contemporary audit practices used in the target environments.

Table 1: Auditable Events

Section	Component	Event	Details
5.1.1	FAU_GEN.1	Start-up and shutdown of the audit functions.	
5.1.2	FAU_GEN.2	None	
5.1.3	FAU_SAR.1	Reading of information from the audit records.	
5.1.4	FAU_SAR.2	Unsuccessful attempts to read information from the audit records.	
5.1.5	FAU_SAR.3	None	

Table 1: Auditable Events

Section	Component	Event	Details
5.1.6	FAU_SEL.1	All modifications to the audit configuration that occur while the audit collection functions are operating.	
5.1.7	FAU_STG.2	None	
5.1.8	FAU_STG.3	Actions taken due to exceeding of a threshold.	
5.1.9	FAU_STG.4	Actions taken due to the audit storage failure.	
5.2.1	FDP_ACC.1	None	
5.2.2	FDP_ACF.1	All requests to perform an operation on an object covered by the SFP.	The identity of the object.
5.2.3	FDP_RIP.2	None	
5.2.4	Note 1	None	
5.3.1	FIA_ATD.1	None	
5.3.2	FIA_SOS.1	Rejection or acceptance by the TSF of any tested secret.	
5.3.3	FIA_UAU.1	All use of the authentication mechanism.	
5.3.4	FIA_UAU.7	None	
5.3.5	FIA_UID.1	All use of the user identification mechanism, including the identity provided *during successful attempts*.	The origin of the attempt (e.g. terminal identification.)
5.3.6	FIA_USB.1	Success and failure of binding user security attributes to a subject (e.g. success and failure to create a subject).	
5.4.1	FMT_MSA.1	All modifications of the values of security attributes.	
5.4.2	FMT_MSA.3	Modifications of the default setting of permissive or restrictive rules. All modifications of the initial value of security attributes.	
5.4.3	FMT_MTD.1	All modifications to the values of TSF data.	
5.4.4	FMT_MTD.1	All modifications to the values of TSF data.	The new value of the TSF data.
5.4.5	FMT_MTD.1	All modifications to the values of TSF data.	The new value of the TSF data.
5.4.6	FMT_MTD.1	All modifications to the values of TSF data.	
5.4.7	FMT_REV.1	All attempts to revoke security attributes.	
5.4.8	FMT_REV.1	All modifications to the values of TSF data.	
5.4.9	FMT_SMR.1	Modifications to the group of users that are part of a role.	

Table 1: Auditable Events

Section	Component	Event	Details
5.4.9	FMT_SMR.1	Every use of the rights of a role. (Additional / Detailed)	The role and the origin of the request.
5.5.1	FPT_AMT.1	Execution of the tests of the underlying machine and the results of the test.	
5.5.2	FPT_RVM.1	None	
5.5.3	FPT_SEP.1	None	
5.5.4	FPT_STM.1	Changes to the time.	

5.1.2 User Identity Association (FAU_GEN.2)

5.1.2.1 The TSF shall be able to associate each auditable event with the identity of the user that caused the event. FAU_GEN.2.1

Application Note: There are some auditable events which may not be associated with a user, such as failed login attempts. It is acceptable that such events do not include a user identity. In the case of failed login attempts it is also acceptable not to record the attempted identity in cases where that attempted identity could be misdirected authentication data; for example when the user may have been out of sync and typed a password in place of a user identifier.

Rationale: O.AUDITING calls for individual accountability (i.e., “TOE users”) whenever security-relevant actions occur. This component requires every auditable event to be associated with an individual user.

5.1.3 Audit Review (FAU_SAR.1)

5.1.3.1 The TSF shall provide authorized administrators with the capability to read all audit information from the audit records: FAU_SAR.1.1

5.1.3.2 The TSF shall provide the audit records in a manner suitable for the user to interpret the information. FAU_SAR.1.2

Application Note: The minimum information which must be provided is the same that which is required to be recorded in 5.1.1.2.

The intent of this requirement is that there exist a tool for administrator be able to access the audit trail in order to assess it. Exactly what manner is provided is an implementation decision, but it needs to be done in a way which allows the administrator to make effective use of the information presented. This requirement is closely tied to 5.1.5 and 5.1.6. It is expected that a single tool will exist within the TSF which will satisfy all of these requirements.

Rationale: This component supports the O.AUDITING and O.MANAGE objectives by providing the administrator with the ability to assess the accountability information accumulated by the TOE.

5.1.4 Restricted Audit Review (FAU_SAR.2)

5.1.4.1 The TSF shall prohibit all users read access to the audit records, except those users that have been granted explicit read-access. FAU_SAR.2.1

Application Note: By default, authorized administrators may be considered to have been granted read access to the audit records. The TSF may provide a mechanism which allows other users to also read audit records.

Rationale: This component supports the O.AUDITING objective by protecting the audit trail from unauthorized access.

5.1.5 Selectable Audit Review (FAU_SAR.3)

5.1.5.1 The TSF shall provide the ability to perform [selection: *searches, sorting*] of audit data based on the following attributes: FAU_SAR.3.1

a) User identity;

b) [assignment: *list of additional attributes that audit selectivity is based upon*]

Application Note: The ST must state the additional attributes that audit selectivity may be based upon (e.g., object identity, type of event), if any.

Rationale: This component supports both the O.AUDITING and O.MANAGE objectives, by providing a means for the administrator to assess the accountability information associated with an individual user.

5.1.6 Selective Audit (FAU_SEL.1)

5.1.6.1 The TSF shall be able to include or exclude auditable events from the set of audited events based on the following attributes: FAU_SEL.1.1

a) User identity;

b) [assignment: *list of additional attributes that audit selectivity is based upon*].

Application Note: The ST must state the additional attributes that audit selectivity may be based upon (e.g., object identity, type of event), if any.

Rationale: This component supports both the O.AUDITING and O.MANAGE objectives, by providing a means for the administrator to assess the accountability information associated with an individual user.

5.1.7 Guarantees of Audit Data Availability (FAU_STG.1)

5.1.7.1 The TSF shall protect the stored audit records from unauthorized deletion. FAU_STG.1.1

5.1.7.2 The TSF shall be able to prevent modifications to the audit records. FAU_STG.1.2

Application Note: On many systems, in order to reduce the performance impact of audit generation, audit records will be temporarily buffered in memory before they are written to disk. In these cases, it is likely that some of these records will be lost if the operation of the TOE is interrupted by hardware or power failures. The developer needs to document what the likely loss will be and show that it has been minimized.

Rationale: This component supports the O.AUDITING objective by protecting the audit trail from tampering, via deletion or modification of records in it. Further it ensures that it is as complete as possible.

5.1.8 Action in Case of Possible Audit Data Loss (FAU_STG.3)

5.1.8.1 The TSF shall generate an alarm to the authorized administrator if the audit trail exceeds [assignment: *pre-defined limit*]. FAU_STG.3.1 / NOTE 3

Application Note: For this component, an “alarm” is to be interpreted as any clear indication to the administrator that the pre-defined limit has been exceeded. The ST author must state the pre-defined limit that triggers generation of the alarm. The limit can be stated as an absolute value, or as a value that represents a percentage of audit trail capacity (e.g., audit trail 75% full). If the limit is adjustable by the authorized administrator, the ST should also incorporate an FMT requirement to manage this function.

Rationale: This component supports the O.AUDITING and O.MANAGE objectives by providing the administrator with a warning that a pending failure due to the exhaustion of space available for audit information.

5.1.9 Prevention of Audit Data Loss (FAU_STG.4)

5.1.9.1 The TSF shall *be able to* prevent auditable events, except those taken by the authorized administrator, and [assignment: *other actions to be taken in case of audit storage failure*] if the audit trail is full. FAU_STG.4.1 / NOTE 5

Application Note: The selection of “preventing” auditable actions if audit storage is exhausted is minimal functionality; providing a range of configurable choices (e.g., ignoring auditable actions and/or changing to a degraded mode) is allowable, as long as “preventing” is one of the choices. If configurable, then FMT_MOF.1 should be incorporated into the ST.

Rationale: This component supports the O.AUDITING and O.MANAGE objectives by providing the audit trail is complete with respect to non-administrative users while providing administrators with the ability to recover from the situation.

5.2 User Data Protection (FDP)

5.2.1 Discretionary Access Control Policy (FDP_ACC.1)

5.2.1.1 The TSF shall enforce the Discretionary Access Control Policy on [assignment: *list of subjects*] acting on the behalf of users, [assignment: *list of named objects*] and all operations among subjects and objects covered by the DAC policy. FDP_ACC.1.1

Application Note: For most systems there is only one type of subject, usually called a process or task, which needs to be specified in the ST.

Named objects are those objects which are used to share information among subjects acting on the behalf of different users, and for which access to the object can be specified by a name or other identity. Any object that meets this criterion but is not controlled by the DAC policy must be justified.

The list of operations covers all operations between the above two lists. It may consist of a sublist for each subject-named object pair. Each operation needs to specify which type of access right is needed to perform the operation; for example read access or write access.

Rationale: This component supports the O.DISCRETIONARY_ACCESS objective by specifying the scope of control for the DAC policy.

5.2.2 Discretionary Access Control Functions (FDP_ACF.1)

5.2.2.1 The TSF shall enforce the Discretionary Access Control Policy to objects based on the following: FDP_ACF.1.1

a) The user identity and group membership(s) associated with a subject; and

b) The following access control attributes associated with an object:

[assignment: List access control attributes. The attributes must provide permission attributes with:

i) the ability to associate allowed or denied operations with one or more user identities;

ii) the ability to associate allowed or denied operations with one or more group identities; and

iii) defaults for allowed or denied operations.]

5.2.2.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: FDP_ACF.1.2

[assignment: a set of rules specifying the Discretionary Access Control policy, where:

i) For each operation there shall be a rule, or rules, that use the permission attributes where the user identity of the subject matches a user identity specified in the access control attributes of the object;

ii) For each operation there shall be a rule, or rules, that use the permission attributes where the group membership of the subject matches a group identity specified in the access control attributes of the object; and

iii) For each operation there shall be a rule, or rules, that use the default permission attributes specified in the access control attributes of the object when neither a user identity or group identity matches.]

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

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

Application Note: A CAPP conformant TOE is required to implement a DAC policy, but the rules which govern the policy may vary between TOEs; those rules need to be specified in the ST. In completing the rule assignment above, the resulting mechanism must be able to specify access rules which apply to at least any single user. This single user may have a special status such as the owner of the object. The mechanism must also support specifying access to the membership of at least any single group. Conformant implementations include self/group/public controls and access control lists.

A DAC policy may cover rules on accessing public objects; i.e., objects which are readable to all authorized users, but which can only be altered by the TSF or authorized

administrators. Specification of these rules should be covered under 5.2.2.3 and 5.2.2.4.

A DAC policy may include exceptions to the basic policy for access by authorized administrators or other forms of special authorization. These rules should be covered under 5.2.2.3.

The ST must list the attributes which are used by the DAC policy for access decisions. These attributes may include permission bits, access control lists, and object ownership.

A single set of access control attributes may be associated with multiple objects, such as all objects stored on a single floppy disk. The association may also be indirectly bound to the object, such as access control attributes being associated with the name of the object rather than directly to the object itself.

Rationale: This component supports the O.DISCRETIONARY_ACCESS objective by defining the rules which will be enforced by the TSF.

5.2.3 Object Residual Information Protection (FDP_RIP.2)

5.2.3.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the allocation of the resource to all objects. FDP_RIP.2

Application Note: This requirement applies to all resources governed by or used by the TSF; it includes resources used to store data and attributes. It also includes the encrypted representation of information.

Clearing the information content of resources on deallocation from objects is sufficient to satisfy this requirement, if unallocated resources will not accumulate new information until they are allocated again.

Rationale: This component supports the O.RESIDUAL_INFORMATION objective.

5.2.4 Subject Residual Information Protection (Note 1)

5.2.4.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the allocation of the resource to all subjects. NOTE 1

Application Note: This requirement applies to all resources governed by or used by the TSF; it includes resources used to store data and attributes. It also includes the encrypted representation of information.

Clearing the information content of resources on deallocation from subjects is sufficient to satisfy this requirement, if unallocated resources will not accumulate new information until they are allocated again.

Rationale: This component supports the O.RESIDUAL_INFORMATION objective.

5.3 Identification and Authentication (FIA)

5.3.1 User Attribute Definition (FIA_ATD.1)

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

a) User Identifier;

b) Group Memberships;

c) Authentication Data;

d) Security-relevant Roles; and

e) [assignment: other user security attributes].

Application Note: The specified attributes are those that are required by the TSF to enforce the DAC policy, the generation of audit records, and proper identification and authentication of users. The user identity must be uniquely associated with a single individual user.

Group membership may be expressed in a number of ways: a list per user specifying to which groups the user belongs, a list per group which includes which users are members, or implicit association between certain user identities and certain groups.

A TOE may have two forms of user and group identities, a text form and a numeric form. In these cases there must be unique mapping between the representations.

Rationale: This component supports the O.AUTHORIZATION and O.DISCRETIONARY_ACCESS objectives by providing the TSF with the information about users needed to enforce the TSP.

5.3.2 Strength of Authentication Data (FIA_SOS.1)

5.3.2.1 The TSF shall provide a mechanism to verify that secrets meet *the following:* FIA_SOS.1

a) For each attempt to use the authentication mechanism, the probability that a random attempt will succeed is less than one in 1,000,000;

b) For multiple attempts to use the authentication mechanism during a one minute period, the probability that a random attempt during that minute will succeed is less than one in 100,000; and

c) Any feedback given during an attempt to use the authentication mechanism will not reduce the probability below the above metrics.

Application Note: The method of authentication is unspecified by the CAPP, but must be specified in a ST. The method which is used must be shown to have low probability that authentication data can be forged or guessed. For example, if a password mechanism is used a set of metrics needs to be specified and may include such things as minimum length of the password, maximum lifetime of a password, and the subjecting of possible passwords to dictionary attacks. The strength of whatever mechanism implemented must be subjected to a strength of function analysis. (See 6.7.2)

Rationale: This component supports the O.AUTHORIZATION objective by providing an authentication mechanism with a reasonable degree of certainty that only authorized users may access the TOE.

5.3.3 Authentication (FIA_UAU.1)

5.3.3.1 The TSF shall allow [assignment: *list of TSF-mediated actions*] on behalf of the user to be performed before the user is authenticated. FIA_UAU.1.1

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

Application Note: The ST must specify the actions which are allowed by an unauthenticated user. The allowed actions should be limited to those things which aid an authorized

user in gaining access to the TOE. This could include help facilities or the ability to send a message to authorized administrators.

Rationale: This component supports the O.AUTHORIZATION objective by specifying what actions unauthenticated users may perform.

5.3.4 Protected Authentication Feedback (FIA_UAU.7)

5.3.4.1 The TSF shall provide only obscured feedback to the user while the authentication is in progress. FIA_UAU.7

Application Note: Obscured feedback implies the TSF does not produce a visible display of any authentication data entered by a user, such as through a keyboard (e.g., echo the password on the terminal). It is acceptable that some indication of progress be returned instead, such as a period returned for each character sent.

Some forms of input, such as card input based batch jobs, may contain human-readable user passwords. The Administrator and User Guidance documentation for the product must explain the risks in placing passwords on such input and must suggest procedures to mitigate that risk.

Rationale: This component supports the O.AUTHORIZATION objective. Individual accountability cannot be maintained if the individual's authentication data, in any form, is compromised.

5.3.5 Identification (FIA_UID.1)

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

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

Application Note: The ST must specify the actions which are allowed to an unidentified user. The allowed actions should be limited to those things which aid an authorized user in gaining access to the TOE. This could include help facilities or the ability to send messages to authorized administrators.

The method of identification is unspecified by this PP, but should be specified in a ST and it should specify how this relates to user identifiers maintained by the TSF.

Rationale: This component supports the O.AUTHORIZATION objective by specifying what actions unidentified users may perform.

5.3.6 User-Subject Binding (FIA_USB.1)

5.3.6.1 The TSF shall associate the *following* user security attributes with subjects acting on the behalf of that user: FIA_USB.1.1 / NOTE 2

a) The user identity which is associated with auditable events;

b) The user identity or identities which are used to enforce the Discretionary Access Control Policy;

c) The group membership or memberships used to enforce the Discretionary Access Control Policy;

d) [assignment: any other user security attributes].

5.3.6.2 The TSF shall enforce the following rules on the initial association of user security attributes with subjects acting on the behalf of a user: NOTE 2

a) [assignment: initial association rules].

5.3.6.3 *The TSF shall enforce the following rules governing changes to the user security attributes associated with subjects acting on the behalf of a user:* NOTE 2

a) [assignment: changing of attributes rules].

Application Note: The DAC policy and audit generation require that each subject acting on the behalf of users have a user identity associated with the subject. This identity is normally the one used at the time of identification to the system.

The DAC policy enforced by the TSF may include provisions for making access decisions based on a user identity which differs from the one used during identification. The ST must state, in 5.3.6.3, how this alternate identity is associated with a subject and justify why the individual user associated with this alternate identity is not compromised by the mechanism used to implement it.

Depending on the TSF's implementation of group membership, the associations between a subject and groups may be explicit at the time of identification or implicit in a relationship between user and group identifiers. The ST must specify this association. Like user identification, an alternate group mechanism may exist, and parallel requirements apply.

Rationale: This component supports the O.DISCRETIONARY_ACCESS and O.AUDITING objectives by binding user identities to subjects acting on their behalf.

5.4 Security Management (FMT)

5.4.1 Management of Object Security Attributes (FMT_MSA.1)

5.4.1.1 The TSF shall enforce the Discretionary Access Control Policy to restrict the ability to modify the access control attributes associated with a named object to [assignment: the *authorized users*]. FMT_MSA.1.1

Application Note: The ST must state the components of the access rights that may be modified, and must state any restrictions that may exist for a type of authorized user and the components of the access rights that the user is allowed to modify.

The ability to modify access rights must be restricted in that a user having access rights to a named object does not have the ability to modify those access rights unless granted the right to do so. This restriction may be explicit, based on the object ownership, or based on a set of object hierarchy rules.

Rationale: This component supports the O.DISCRETIONARY_ACCESS objective by providing the means by which the security attributes of objects are managed by a site.

5.4.2 Static Attribute Initialization (FMT_MSA.3)

5.4.2.1 The TSF shall enforce the Discretionary Access Control Policy to provide restrictive default values for security attributes that are used to enforce the Discretionary Access Control Policy. FMT_MSA.3.1

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

Application Note: A CAPP-conformant TOE must provide protection by default for all objects at creation time. This may be done through the enforcing of a restrictive default access control on newly created objects or by requiring the user to explicitly specify the desired access controls on the object at its creation. In either case, there shall be no window of vulnerability through which unauthorized access may be gained to newly created objects.

Rationale: This component supports the O.DISCRETIONARY_ACCESS objective by requiring that objects are properly protected starting from the instant that they are created.

5.4.3 Management of the Audit Trail (FMT_MTD.1)

5.4.3.1 The TSF shall restrict the ability to create, delete, and clear the audit trail to authorized administrators. FMT_MTD.1.1

Application Note: The selection of “create, delete, and clear” functions for audit trail management reflect common management functions. These functions should be considered generic; any other audit administration functions that are critical to the management of a particular audit mechanism implementation should be specified in the ST.

Rationale: The component supports the O.AUDITING and O.MANAGE objectives by ensuring that the accountability information is not compromised by destruction of the audit trail.

5.4.4 Management of Audited Events (FMT_MTD.1)

5.4.4.1 The TSF shall restrict the ability to modify or observe the set of audited events to authorized administrators. FMT_MTD.1.1

Application Note: The set of audited events are the subset of auditable events which will be audited by the TSF. The term set is used loosely here and refers to the total collection of possible ways to control which audit records get generated; this could be by type of record, identity of user, identity of object, etc.

It is an important aspect of audit that users not be able to effect which of their actions are audited, and therefore must not have control over or knowledge of the selection of an event for auditing.

Rationale: This component supports the O.AUDITING and O.MANAGE objectives by providing the administrator with the ability to control the degree to which accountability is generated.

5.4.5 Management of User Attributes (FMT_MTD.1)

5.4.5.1 The TSF shall restrict the ability to initialize and modify the user security attributes, other than authentication data, to authorized administrators. FMT_MTD.1.1

Application Note: This component only applies to security attributes which are used to maintain the TSP. Other user attributes may be specified in the ST, but control of those attributes are not within the scope of the CAPP.

Rationale: This component supports the O.MANAGE objective by providing the administrator with the means to manage who are authorized users and what attributes are associated with each user.

5.4.6 Management of Authentication Data (FMT_MTD.1)

5.4.6.1 The TSF shall restrict the ability to initialize the authentication data to authorized administrators. FMT_MTD.1.1

5.4.6.2 The TSF shall restrict the ability to modify the authentication data to the following: FMT_MTD.1.1

a) authorized administrators; and

b) users authorized to modify their own authentication data

Application Note: User authentication data refers to information that users must provide to authenticate themselves to the TSF. Examples include passwords, personal identification numbers, and fingerprint profiles. User authentication data does not include the users identity. The ST must specify the authentication mechanism that makes use of the user authentication data to verify a user's identity.

This component does not require that any user be authorized to modify their own authentication information; it only states that it is permissible. It is not necessary that requests to modify authentication data require reauthentication of the requester's identity at the time of the request.

Rationale: This component supports the O.AUTHORIZATION and O.MANAGE objectives by ensuring integrity and confidentiality of authentication data.

5.4.7 Revocation of User Attributes (FMT_REV.1)

5.4.7.1 The TSF shall restrict the ability to revoke security attributes associated with the users within the TSC to authorized administrators. FMT_REV.1.1

5.4.7.2 The TSF shall enforce the rules: FMT_REV.1.2

a) The immediate revocation of security-relevant authorizations; and

b) [assignment: list of other revocation rules concerning users].

Application Note: Many security-relevant authorizations could have serious consequences if misused, so an immediate revocation method must exist, although it need not be the usual method (e.g., The usual method may be editing the trusted users profile, but the change doesn't take effect until the user logs off and logs back on. The method for immediate revocation might be to edit the trusted users profile and "force" the trusted user to log off.). The immediate method must be specified in the ST and in administrator guidance. In a distributed environment the developer must provide a description of how the "immediate" aspect of this requirement is met.

Rationale: This component supports the O.MANAGE objective by controlling access to data and functions which are not generally available to all users.

5.4.8 Revocation of Object Attributes (FMT_REV.1)

The TSF shall restrict the ability to revoke security attributes associated with objects within the TSC to users authorized to modify the security attributes by the Discretionary Access Control policy.[FMT_REV.1.1]

5.4.8.1 The TSF shall enforce the rules: [FMT_REV.]

a) The access rights associated with an object shall be enforced when an access check is made; and

b) [assignment: list of other revocation rules concerning objects].

Application Note: The DAC policy may include immediate revocation (e.g., Multics immediately revokes access to segments) or delayed revocation (e.g., most UNIX systems do not revoke access to already opened files). The DAC access rights are considered to have been revoked when all subsequent access control decisions by the TSF use the new access control information. It is not required that every operation on an object make an explicit access control decision as long as a previous access control decision was made to permit that operation. It is sufficient that the developer clearly documents in guidance documentation how revocation is enforced.

Rationale: This component supports the O.DISCRETIONARY_ACCESS objective by providing that specified access control attributes are enforced at some fixed point in time.

5.4.9 Security Management Roles (FMT_SMR.1)

5.4.9.1 The TSF shall maintain the roles: [FMT_SMR.1.1]

a) authorized administrator;

b) users authorized by the Discretionary Access Control Policy to modify object security attributes;

c) users authorized to modify their own authentication data; and

d) [assignment: other roles].

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

Application Note: A CAPP-conformant TOE only needs to support a single administrative role, referred to as the authorized administrator. If a TOE implements multiple independent roles, the ST should refine the use of the term authorized administrators to specify which roles fulfill which requirements.

The CAPP specifies a number of functions which are required of or restricted to an authorized administrator, but there may be additional functions which are specific to the TOE. This would include any additional function which would undermined the proper operation of the TSF. Examples of functions include: ability to access certain system resources like tape drives or vector processors, ability to manipulate the printer queues, and ability to run real-time programs.

Rationale: This component supports the O.MANAGE objective.

5.5 Protection of the TOE Security Functions (FPT)

5.5.1 Abstract Machine Testing (FPT_AMT.1)

5.5.1.1 The TSF shall run a suite of tests [selection: *during initial start-up, periodically during normal operation, or at the request of an authorized administrator*] to demonstrate the correct operation of the security assumptions provided by the abstract machine that underlies the TSF. FPT_AMT.1.1

Application Note: In general this component refers to the proper operation of the hardware platform on which a TOE is running. The test suite needs to cover only aspects of the hardware on which the TSF relies to implement required functions, including domain separation. If a failure of some aspect of the hardware would not result in the TSF compromising the functions it performs, then testing of that aspect is not required.

Rationale: This component supports the O.ENFORCEMENT objective by demonstrating that the underlying mechanisms are working as expected.

5.5.2 Reference Mediation (FPT_RVM.1)

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

Application Note: This element does not imply that there must be a reference monitor. Rather this requires that the TSF validates all actions between subjects and objects that require policy enforcement.

Rationale: This component supports O.ENFORCEMENT objective by ensuring that the TSP is not being bypassed.

5.5.3 Domain Separation (FPT_SEP.1)

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

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

Application Note: This component does not imply a particular implementation of a TOE. The implementation needs to exhibit properties that the code and the data upon which TSF relies are not alterable in ways that would compromise the TSF and that observation of TSF data would not result in failure of the TSF to perform its job. This could be done either by hardware mechanisms or hardware architecture. Possible implementations include multi-state CPU's which support multiple task spaces and independent nodes within a distributed architecture.

The second element can also be met in a variety of ways also, including CPU support for separate address spaces, separate hardware components, or entirely in software. The latter is likely in layered application such as a graphic user interface system which maintains separate subjects.

Rationale: This component supports O.ENFORCEMENT objectives by ensuring that a TSF exists within the TOE and that it can reliably carry out its functions.

5.5.4 Reliable Time Stamps (FPT_STM.1)

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

Application Note: The generation of audit records depends on having a correct date and time. The ST needs to specify the degree of accuracy that must be maintained in order to maintain useful information for audit records.

Rationale: This component supports the O.AUDITING objective by ensuring that accountability information is accurate.

6.0 Assurance Requirements

This chapter defines the assurance requirements for the TOE. Assurance requirement components are Evaluation Assurance Level (EAL) 3, with no augmentation, from part 3 of the CC.

6.1 Configuration Management (ACM)

6.1.1 Authorization Controls (ACM_CAP.3)

6.1.1.1D The developer shall provide a reference for the TOE. ACM_CAP.3.1D

6.1.1.2D The developer shall use a CM system. ACM_CAP.3.2D

6.1.1.3D The developer shall provide CM documentation. ACM_CAP.3.3D

6.1.1.1C The reference for the TOE shall be unique to each version of the TOE. ACM_CAP.3.1C

6.1.1.2C The TOE shall be labelled with its reference. ACM_CAP.3.2C

6.1.1.3C The CM documentation shall include a configuration list and a CM plan. ACM_CAP.3.3C

6.1.1.4C The configuration list shall describe the configuration items that comprise the TOE. ACM_CAP.3.4C

6.1.1.5C The CM documentation shall describe the method used to uniquely identify the configuration items. ACM_CAP.3.5C

6.1.1.6C The CM system shall uniquely identify all configuration items. ACM_CAP.3.6C

6.1.1.7C The CM plan shall describe how the CM system is used. ACM_CAP.3.7C

6.1.1.8C The evidence shall demonstrate that the CM system is operating in accordance with the CM plan. ACM_CAP.3.8C

6.1.1.9C The CM documentation shall provide evidence that all configuration items have been and are being effectively maintained under the CM system. ACM_CAP.3.9C

6.1.1.10C The CM system shall provide measures such that only authorized changes are made to the configuration items. ACM_CAP.3.10C

6.1.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. ACM_CAP.3.1E

Application Note: This component provides for three things. First it requires that the TOE is identifiable by a customer, using things such as version and part numbers, to ensure that the proper thing has been installed. Second it requires that the pieces used to produce the TOE are identified; the scope required is covered in 6.1.2. And third it requires that the production of the TOE be done in a controlled manner.

6.1.2 Coverage (ACM_SCP.1)

6.1.2.1D The developer shall provide CM documentation. ACM_SCP.1.1D

6.1.2.1C The CM documentation shall show that the CM system, as a minimum, tracks the following: the TOE implementation representation, design documentation, test documentation, user documentation, administrator documentation, and CM documentation. ACM_SCP.1.1C

6.1.2.2C The CM documentation shall describe how configuration items are tracked by the CM system. ACM_SCP.1.2C

6.1.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. ACM_SCP.1.1E

6.2 Delivery and Operation (ADO)

6.2.1 Delivery Procedures (ADO_DEL.1)

6.2.1.1D The developer shall document procedures for delivery of the TOE or parts of it to the user. ADO_DEL.1.1D

6.2.1.2D The developer shall use the delivery procedures. ADO_DEL.1.2D

6.2.1.1C The delivery documentation shall describe all procedures that are necessary to maintain security when distributing versions of the TOE to a user's site. ADO_DEL.1.1C

6.2.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. ADO_DEL.1.1E

Application Note: The delivery procedures for a CAPP conformant TOE can vary greatly and can range from a shrink wrapped box from a retail outlet to delivery by a field engineer. As such, there may be opportunities for third parties to tamper with the TOE delivery process. In these cases the developer should provide procedures or mechanisms to mitigate the threat.

6.2.2 Installation, Generation, and Start-up Procedures (ADO_IGS.1)

6.2.2.1D The developer shall document procedures necessary for the secure installation, generation, and start-up of the TOE. ADO_IGS.1.1D

6.2.2.1C The documentation shall describe the steps necessary for secure installation, generation, and start-up of the TOE. ADO_IGS.1.1C

6.2.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. ADO_IGS.1.1E

6.2.2.2E The evaluator shall determine that the installation, generation, and start-up procedures result in a secure configuration. ADO_IGS.1.2E

Application Note: The required documentation depends on the way in which a TOE is generated and installed. For example the generation of a TOE from source code may be done at the development site, in which case the required documentation would be considered part of the design documentation. On the other hand, if some part of the TOE generation is done by the TOE administrator, it would be part of the administrative guidance. Similar circumstances could also apply to both installation and start-up procedures.

6.3 Development (ADV)

6.3.1 Functional Specification (ADV_FSP.1)

6.3.1.1D The developer shall provide a functional specification. ADV_FSP.1.1D

6.3.1.1C The functional specification shall describe the TSF and its external interfaces using an informal style. ADV_FSP.1.1C

6.3.1.2C The functional specification shall be internally consistent. ADV_FSP.1.2C

6.3.1.3C 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. ADV_FSP.1.3C

6.3.1.4C The functional specification shall completely represent the TSF. ADV_FSP.1.4C

6.3.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. ADV_FSP.1.1E

6.3.1.2E The evaluator shall determine that the functional specification is an accurate and complete instantiation of the TOE security functional requirements. ADV_FSP.1.2E

Application Note: This components requires that the design documentation includes a complete external description of the TSF. In particular it needs to address the mechanisms which are used to meet the functional requirements of the CAPP. Other areas need to be addressed to the degree that they affect the functional requirements.

6.3.2 High-Level Design (ADV_HLD.2)

6.3.2.1D The developer shall provide the high-level design of the TSF. ADV_HLD.2.1D

6.3.2.1C The presentation of the high-level design shall be informal. ADV_HLD.2.1C

6.3.2.2C The high-level design shall be internally consistent. ADV_HLD.2.2C

6.3.2.3C The high-level design shall describe the structure of the TSF in terms of subsystems. ADV_HLD.2.3C

6.3.2.4C The high-level design shall describe the security functionality provided by each subsystem of the TSF. ADV_HLD.2.4C

6.3.2.5C The high-level design shall identify any underlying hardware, firmware, and/or software required by the TSF with a presentation of the functions provided by the supporting protection mechanisms implemented in that hardware, firmware, or software. ADV_HLD.2.5C

6.3.2.6C The high-level design shall identify all interfaces to the subsystems of the TSF. ADV_HLD.2.6C

6.3.2.7C The high-level design shall identify which of the interfaces to the subsystems of the TSF are externally visible. ADV_HLD.2.7C

6.3.2.8C The high-level design shall describe the purpose and method of use of all interfaces to the subsystems of the TSF, providing details of effects, exceptions and error messages, as appropriate. ADV_HLD.2.8C

6.3.2.9C The high-level design shall describe the separation of the TSF into TSP-enforcing and other subsystems. ADV_HLD.2.9C

6.3.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. ADV_HLD.2.1E

6.3.2.2E The evaluator shall determine that the high-level design is an accurate and complete instantiation of the TOE security functional requirements. ADV_HLD.2.2E

Application Note: This component requires that the design documentation include a breakdown of the TSF at a very coarse grain. Both the developer and evaluator need to carefully choose how a “subsystem” is defined for a particular TOE. There must be a balance between subsystems being too large that it is difficult to understand the functions of any single subsystem and subsystems that are so small that how they fit into the system as a whole is difficult to understand. If different pieces of the TSF are developed or maintained by different groups of developers, that can aid in making those choices.

Furthermore, it must be noted that the presentation need only be informal. This means that the interfaces between subsystems need to be presented in general terms of how they interact, not to the level of presenting a programming interface specification between them.

6.3.3 Correspondence Demonstration (ADV_RCR.1)

6.3.3.1D The developer shall provide an analysis of correspondence between all adjacent pairs of TSF representations that are provided. ADV_RCR.1.1D

6.3.3.1C For each adjacent pair of provided TSF representations, the analysis shall demonstrate that all relevant security functionality of the more abstract TSF representation is correctly and completely refined in the less abstract TSF representation. ADV_RCR.1.1C

6.3.3.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. ADV_RCR.1.1E

Application Note: For the CAPP, this ensures that the functional specification and high-level design are consistent with each other.

6.4 Guidance Documents (AGD)

6.4.1 Administrator Guidance (AGD_ADM.1)

6.4.1.1D The developer shall provide administrator guidance addressed to system administrative personnel. AGD_ADM.1.1D

6.4.1.1C The administrator guidance shall describe the administrative functions and interfaces available to the administrator of the TOE. AGD_ADM.1.1C

6.4.1.2C The administrator guidance shall describe how to administer the TOE in a secure manner. AGD_ADM.1.2C

6.4.1.3C The administrator guidance shall contain warnings about functions and privileges that should be controlled in a secure processing environment. AGD_ADM.1.3C

6.4.1.4C The administrator guidance shall describe all assumptions regarding user behavior that are relevant to secure operation of the TOE. AGD_ADM.1.4C

6.4.1.5C The administrator guidance shall describe all security parameters under the control of the administrator, indicating secure values as appropriate. AGD_ADM.1.5C

6.4.1.6C The administrator guidance shall describe each type of security-relevant event relative to the administrative functions that need to be performed, including changing the security characteristics of entities under the control of the TSF. AGD_ADM.1.6C

6.4.1.7C The administrator guidance shall be consistent with all other documents supplied for evaluation. AGD_ADM.1.7C

6.4.1.8C The administrator guidance shall describe all security requirements on the IT environment that are relevant to the administrator. AGD_ADM.1.8C

6.4.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. AGD_ADM.1.1E

Application Note: The content required by this component is quite comprehensive and broadly stated; in particular the content needs to address any of the mechanisms and functions provided to an administrator to meet the functional requirements of the CAPP. It should also contain warnings about actions that may typically be done by administrators which should not be done on this specific TOE. This may include activating certain features or installing certain software which would compromise the TSF.

6.4.2 User Guidance (AGD_USR.1)

6.4.2.1D The developer shall provide user guidance. AGD_USR.1.1D

6.4.2.1C The user guidance shall describe the functions and interfaces available to the non-administrative users of the TOE. AGD_USR.1.1C

6.4.2.2C The user guidance shall describe the use of user-accessible security functions provided by the TOE. AGD_USR.1.2C

6.4.2.3C The user guidance shall contain warnings about user-accessible functions and privileges that should be controlled in a secure processing environment. AGD_USR.1.3C

6.4.2.4C The user guidance shall clearly present all user responsibilities necessary for secure operation of the TOE, including those related to assumptions regarding user behavior found in the statement of TOE security environment. AGD_USR.1.4C

6.4.2.5C The user guidance shall be consistent with all other documentation supplied for evaluation. AGD_USR.1.5C

6.4.2.6C The user guidance shall describe all security requirements on the IT environment that are relevant to the user. AGD_USR.1.6C

6.4.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. AGD_USR.1.1E

Application Note: The content required by this component is quite comprehensive and broadly stated; in particular the content needs to address any of the mechanisms and functions provided to a user to meet the functional requirements of the CAPP. It should also contain warnings about certain actions that may typically be done by users which should not be done on this specific TOE.

6.5 Life Cycle Support (ALC)

6.5.1 Identification of Security Measures (ALC_DVS.1)

6.5.1.1D The developer shall produce development security documentation. ALC_DVS.1.1D

6.5.1.1C The development security documentation shall describe all the physical, procedural, personnel, and other security measures that are necessary to protect the confidentiality and integrity of the TOE design and implementation in its development environment. ALC_DVS.1.1C

6.5.1.2C The development security documentation shall provide evidence that these security measures are followed during the development and maintenance of the TOE. ALC_DVS.1.2C

6.5.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. ALC_DVS.1.1E

6.5.1.2E The evaluator shall confirm that the security measures are being applied. ALC_DVS.1.2E

Application Note: For the CAPP, this is really an extension of the configuration management system requirements to reduce the chance that the TSF is subverted by outsiders during development.

6.6 Security Testing (ATE)

6.6.1 Coverage (ATE_COV.2)

6.6.1.1D The developer shall provide an analysis of the test coverage. ATE_COV.2.1D

6.6.1.1C The analysis of the test coverage shall demonstrate the correspondence between the tests identified in the test documentation and the TSF as described in the functional specification. ATE_COV.2.1C

6.6.1.2C The analysis of the test coverage shall demonstrate that the correspondence between the TSF as described in the functional specification and the tests identified in the test documentation is complete. ATR_COV.2.2C

6.6.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. ATE_COV.2.1E

6.6.2 Depth (ATE_DPT.1)

6.6.2.1D The developer shall provide the analysis of the depth of testing. ATE_DPT.1.1D

6.6.2.1C The depth analysis shall demonstrate that the tests identified in the test documentation are sufficient to demonstrate that the TSF operates in accordance with its high-level design. ATE_DPT.1.1C

6.6.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. ATE_DPT.1.1E

Application Note: While the high-level design is to be used as the basis for testing, it is not required that internal interfaces between subsystems be tested.

6.6.3 Functional Testing (ATE_FUN.1)

6.6.3.1D The developer shall test the TSF and document the results. ATE_FUN.1.1D

6.6.3.2D The developer shall provide test documentation. ATE_FUN.1.2D

6.6.3.1C The test documentation shall consist of test plans, test procedure descriptions, expected test results and actual test results. ATE_FUN.1.1C

6.6.3.2C The test plans shall identify the security functions to be tested and describe the goal of the tests to be performed. ATE_FUN.1.2C

6.6.3.3C The test procedure descriptions shall identify the tests to be performed and describe the scenarios for testing each security function. These scenarios shall include any ordering dependencies on the results of other tests. ATE_FUN.1.3C

6.6.3.4C The expected test results shall show the anticipated outputs from a successful execution of the tests. ATE_FUN.1.4C

6.6.3.5C The test results from the developer execution of the tests shall demonstrate that each tested security function behaved as specified. ATE_FUN.1.5C

6.6.3.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. ATE_FUN.1.1E

6.6.4 Independent Testing (ATE_IND.2)

6.6.4.1D The developer shall provide the TOE for testing. ATE_IND.2.1D

6.6.4.1C The TOE shall be suitable for testing. ATE_IND.2.1C

6.6.4.2C The developer shall provide an equivalent set of resources to those that were used in the developer's functional testing of the TSF. ATE_IND.2.2C

6.6.4.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. ATE_IND.2.1.E

6.6.4.2E The evaluator shall test a subset of the TSF as appropriate to confirm that the TOE operates as specified. ATE_IND.2.2E

6.6.4.3E The evaluator shall execute a sample of tests in the test documentation to verify the developer test results. ATE_IND.2.3E

Application Note: The choice of the subset tested and sample of tests executed by the evaluator is entirely at the discretion of the evaluator.

6.7 Vulnerability Assessment (AVA)

6.7.1 Examination of Guidance (AVA_MSU.1)

6.7.1.1D The developer shall provide guidance documentation. AVA_MSU.1.1D

6.7.1.1C The guidance documentation shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation. AVA_MSU.1.1C

6.7.1.2C The guidance documentation shall be complete, clear, consistent and reasonable. AVA_MSU.1.2C

6.7.1.3C The guidance documentation shall list all assumptions about the intended environment. AVA_MSU.1.3C

6.7.1.4C The guidance documentation shall list all requirements for external security measures (including external procedural, physical and personnel controls). AVA_MSU.1.4C

6.7.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. AVA_MSU.1.1E

6.7.1.2E The evaluator shall repeat all configuration and installation procedures to confirm that the TOE can be configured and used securely using only the supplied guidance documentation. AVA_MSU.1.2E

6.7.1.3E The evaluator shall determine that the use of the guidance documentation allows all insecure states to be detected. AVA_MSU.1.3E

Application Note: This requirement is can be approached as testing by the evaluator to ensure that the guidance documents are correct. The content elements primarily reinforce the guidance requirements themselves.

6.7.2 Strength of TOE Security Function Evaluation (AVA_SOF.1)

6.7.2.1D The developer shall perform a strength of TOE security function analysis for each mechanism identified in the ST as having a strength of TOE security function claim. AVA_SOF.1.1D

6.7.2.1C For each mechanism with a strength of TOE security function claim the strength of TOE security function analysis shall show that it meets or exceeds the minimum strength level defined in the PP/ST. AVA_SOF.1.1C

6.7.2.2C For each mechanism with a specific strength of TOE security function claim the strength of TOE security function analysis shall show that it meets or exceeds the specific strength of function metric defined in the PP/ST. AVA_SOF.1.2C

6.7.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. AVA_SOF.1.1E

6.7.2.2E The evaluator shall confirm that the strength claims are correct. AVA_SOF.1.2E

Application Note: For the CAPP, the requirement applies to the authentication mechanism which is used as described in 5.3.2.

6.7.3 Developer Vulnerability Analysis (AVA_VLA.1)

6.7.3.1D The developer shall perform and document an analysis of the TOE deliverables searching for obvious ways in which a user can violate the TSP. AVA_VLA.1.1D

6.7.3.2D The developer shall document the disposition of obvious vulnerabilities. AVA_VLA.1.2D

6.7.3.1C The documentation shall show, for all identified vulnerabilities, that the vulnerability cannot be exploited in the intended environment for the TOE. AVA_VLA.1.1C

6.7.3.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence. AVA_VLA.1.1E

6.7.3.2E The evaluator shall conduct penetration testing, building on the developer vulnerability analysis, to ensure obvious vulnerabilities have been addressed. AVA_VLA.1.2E

Application Note: The evaluator should consider the following with respect to the search for obvious flaws:

a) Dependencies among functional components and potential inconsistencies in strength of function among interdependent functions;

b) Potential inconsistencies between the TSP and the functional specification;

c) Potential gaps or inconsistencies in the HLD, and potentially invalid assumptions about supporting hardware, firmware, and/or software required by the TSF;

d) Potential gaps in the administrator guidance that enable the administrator to fail (a) to make effective use of TSF functions, (b) to understand or take actions that need to be performed, (c) to avoid unintended interactions among security functions, and (d) to install and/or configure the TOE correctly. In particular, failure to describe all the security parameters under the administrator's control and the effects of settings of (interacting combinations of) those parameters;

e) Potential gaps in the user guidance that enable the user to fail to control functions and privileges as required to maintain a secure processing environment. Potential presence in the user guidance of information that facilitates exploitation of vulnerabilities;

f) Open literature (e.g., CERT advisories, bug-traq mailing list) which may contain information on vulnerabilities on the TSF should be consulted.

7.0 Rationale

This chapter provides the rationale for the selection, creation, and use of the security policies, objectives, and components. Section 7.1 provides the rationale for the existence of the security objectives based upon the stated security policies while Section 7.2 provides the lower-level rationale for the existence of functional and assurance components based upon the stated security objectives. Section 7.2 provides an analysis that maps given security objectives to components as well as mapping given components to security objectives. In providing a mapping in both directions for the components and objectives, assurance is gained that the objectives were entirely met. This is further detailed in Section 7.2.

In addition to providing a complete rationale, chapters 5 and 6 also provide the necessary application notes needed to understand how a TOE must meet the stated security objectives. These application notes provide additional information about a particular family/component/element that a developer or evaluator may need in order to fully understand how the component is to be applied.

7.1 Security Objectives Rationale

This section provides a rationale for the existence of each threat, policy statement, security objective, and component that comprise the protection profile.

7.1.1 Complete Coverage - Threats

The TOE security objectives have been derived exclusively from statements of organizational security policy, and therefore, there are no explicitly defined threats countered by this profile.

7.1.2 Complete Coverage - Policy

This section provides evidence demonstrating coverage of the Organizational Security Policy by both the IT and Non-IT security objectives. The following table shows this objective to policy mapping, and the table is followed by a discussion of the coverage for each Security Policy.

Organizational Security Policy	Security Objectives
P.AUTHORIZED_USERS	O.AUTHORIZATION O.MANAGE O.ENFORCEMENT
P.NEED_TO_KNOW	O.DISCRETIONARY_ACCESS O.RESIDUAL_INFORMATION O.MANAGE O.ENFORCEMENT
P.ACCOUNTABILITY	O.AUDITING O.MANAGE O.ENFORCEMENT

The following discussion provides detailed evidence of coverage for each statement of organizational security policy:

P.AUTHORIZED_USERS

Only those users who have been authorized to access the information within the system may access the system.

This policy is implemented by the O.AUTHORIZATION objective. The O.MANAGE supports this policy by requiring authorized administrators to be able to manage the functions and O.ENFORCEMENT ensures that functions are invoked and operate correctly.

P.NEED_TO_KNOW

The system must limit the access to, modification of, and destruction of the information in protected resources to those authorized users which have a "need to know" for that information.

This policy is implemented by the O.DISCRETIONARY_ACCESS objective. The O.RESIDUAL_INFORMATION objective ensures that information will not given to users which do not have a need to know, when resources are reused. The O.MANAGE supports this policy by requiring authorized administrator be able to manage the functions and O.ENFORCEMENT ensures that functions are invoked and operate correctly.

P.ACCOUNTABILITY

The users of the system shall be held accountable for their actions within the system.

This policy is implemented by the O.AUDITING objective by requiring that actions are recorded in an audit trail. The O.MANAGE supports this policy by requiring authorized administrator be able to manage the functions and O.ENFORCEMENT ensures that functions are invoked and operate correctly.

7.1.3 Complete Coverage - Environmental Assumptions

This section provides evidence demonstrating coverage of the Non-IT security objectives by the environmental assumptions. The following table shows this assumption to objective mapping.

Non IT Security Objectives	Environmental Assumptions
O.INSTALL	A.MANAGE A.NO_EVIL_ADM A.PEER
O.PHYSICAL	A.LOCATE A.PROTECT A.CONNECT
O.CREDEN	A.COOP

7.2 Security Requirements Rationale

This section provides evidence supporting the combined internal consistency and completeness of the functional components that comprise the CAPP.

7.2.1 Internal Consistency of Requirements

This section describes the mutual support and internal consistency of the components selected for this profile. These properties are discussed for both functional and assurance components.

The functional components were selected from pre-defined CC components. The use of component refinement was accomplished in accordance with CC guidelines. An additional component was included to clarify the relationship of objects and security attributes.

Assignment, selection, and refinement operations were carried out among components using consistent computer security terminology. This helps to avoid the ambiguity associated with interpretations of meanings of terms between related components.

Multiple instantiation of identical or hierarchically-related components was used to clearly state the required functionality that must exist in a TOE conformant with this profile.

7.2.2 Complete Coverage - Objectives

This section demonstrates that the functional components selected for this profile provide complete coverage of the defined security objectives. The mapping of components to security objectives is

depicted in the following table.

Security Objective	Functional Component
O.AUTHORIZATION	5.3.1 User Attribute Definition (FIA_ATD.1) 5.3.2 Strength of Authentication Data (FIA_SOS.1) 5.3.3 Authentication (FIA_UAU.1) 5.3.4 Protected Authentication Feedback (FIA_UAU.7) 5.3.5 Identification (FIA_UID.1) 5.4.6 Management of Authentication Data (FMT_MTD.1)
O.DISCRETIONARY_ACCESS	5.2.1 Discretionary Access Control Policy (FDP_ACC.1) 5.2.2 Discretionary Access Control Functions (FDP_ACF.1) 5.3.1 User Attribute Definition (FIA_ATD.1) 5.3.6 User-Subject Binding (FIA_USB.1) 5.4.1 Management of Object Security Attributes (FMT_MSA.1) 5.4.2 Static Attribute Initialization (FMT_MSA.3) 5.4.8 Revocation of Object Attributes (FMT_REV.1)
O.AUDITING	5.1.1 Audit Data Generation (FAU_GEN.1) 5.1.2 User Identity Association (FAU_GEN.2) 5.1.3 Audit Review (FAU_SAR.1) 5.1.4 Restricted Audit Review (FAU_SAR.2) 5.1.5 Selectable Audit Review (FAU_SAR.3) 5.1.6 Selective Audit (FAU_SEL.1) 5.1.7 Guarantees of Audit Data Availability (FAU_STG.1) 5.1.8 Action in Case of Possible Audit Data Loss (FAU_STG.3) 5.1.9 Prevention of Audit Data Loss (FAU_STG.4) 5.3.6 User-Subject Binding (FIA_USB.1) 5.4.3 Management of the Audit Trail (FMT_MTD.1) 5.4.4 Management of Audited Events (FMT_MTD.1) 5.5.4 Reliable Time Stamps (FPT_STM.1)
O.RESIDUAL_INFORMATION	5.2.3 Object Residual Information Protection (FDP_RIP.2) 5.2.4 Subject Residual Information Protection (Note 1)
O.MANAGE	5.1.3 Audit Review (FAU_SAR.1) 5.1.5 Selectable Audit Review (FAU_SAR.3) 5.1.6 Selective Audit (FAU_SEL.1) 5.1.8 Action in Case of Possible Audit Data Loss (FAU_STG.3) 5.1.9 Prevention of Audit Data Loss (FAU_STG.4) 5.4.3 Management of the Audit Trail (FMT_MTD.1) 5.4.4 Management of Audited Events (FMT_MTD.1) 5.4.5 Management of User Attributes (FMT_MTD.1) 5.4.6 Management of Authentication Data (FMT_MTD.1) 5.4.7 Revocation of User Attributes (FMT_REV.1) 5.4.9 Security Management Roles (FMT_SMR.1)
O.ENFORCEMENT	5.5.1 Abstract Machine Testing (FPT_AMT.1) 5.5.2 Reference Mediation (FPT_RVM.1) 5.5.3 Domain Separation (FPT_SEP.1)

The following discussion provides detailed evidence of coverage for each security objective:

O.AUTHORIZATION

The TSF must ensure that only authorized users gain access to the TOE and its resources.

Users authorized to access the TOE are defined using an identification and authentication process [5.3.5, 5.3.3]. To ensure authorized access to the TOE, authentication data is protected [5.3.1, 5.3.4, 5.4.6]. The strength of the authentication mechanism must be sufficient to ensure unauthorized users to easily pose as authorized users [5.3.2].

O.DISCRETIONARY_ACCESS

The TSF must control accessed to resources based on identity of users. The TSF must allow authorized users to specify which resources may be accessed by which users.

Discretionary access control must have a defined scope of control [5.2.1]. The rules of the DAC policy must be defined [5.2.2]. The security attributes of objects used to enforce the DAC policy must be defined. The security attributes of subjects used to enforce the DAC policy must be defined [5.3.1, 5.3.6]. Authorized users must be able to control who has access to objects [5.4.1] and be able to revoke that access [5.4.8]. Protection of named objects must be continuous, starting from object creation [5.4.2].

O.AUDITING

The TSF must record the security relevant actions of users of the TOE. The TSF must present this information to authorized administrators.

Security-relevant actions must be defined, auditable [5.1.1], and capable of being associated with individual users [5.1.2, 5.3.6]. The audit trail must be protected so that only authorized users may access it [5.1.4]. The TSF must provide the capability to audit the actions of an individual user [5.1.5, 5.1.6, 5.3.6]. The audit trail must be complete [5.1.7, 5.1.9]. The time stamp associated must be reliable [5.5.4]. An authorized administrator must be able to review [5.1.3] and manage [5.1.8, 5.4.4, 5.4.4] the audit trail.

O.RESIDUAL_INFORMATION

The TSF must ensure that any information contained in a protected resource is not released when the resource is recycled.

Residual information associated with defined objects in the TOE must be purged prior to the reuse of the object containing the residual information [5.2.3].

O.MANAGE

The TSF must provide all the functions and facilities necessary to support the authorized administrators that are responsible for the management of TOE security.

The TSF must provide for an authorized administrator to manage the TOE [5.4.9]. The administrator must be able to administer user accounts [5.4.5, 5.4.6, 5.4.7]. The administrator must be able to review manage the audit trail [5.1.3, 5.1.5, 5.1.6, 5.1.8, 5.1.9, 5.4.3, 5.4.4].

O.ENFORCEMENT

The TSF must be designed and implemented in a manner which ensures that the organizational policies are enforced in the target environment.

The TSF must make and enforce the decisions of the TSP [5.5.2]. It must be protected from interference that would prevent it from performing its functions [5.5.3]. Additionally, the TOE must provide the capability to demonstrate correct operation of the TSF's underlying abstract machine [5.5.1]. The correctness of this objective is further met through the assurance requirements defined in this PP.

This objective provides global support to other security objectives for the TOE by protecting the parts

of the TOE which implement policies and ensures that policies are enforced.

7.3 Dependencies

The following table shows the dependencies which exist. A box with an X in it indicates a dependency which has been satisfied. A box with an O in it indicates an optional dependency where one of the options has been satisfied.

Section	CC Identifier	FAU_GEN.1	FAU_SAR.1	FAU_STG.1	FDP_ACC.1	FDP_ACF.1	FDP_IFC.1	FDP_IFF.1	FIA_ATD.1	FIA_UAU.1	FIA_UID.1	FMT_MSA.1	FMT_MSA.3	FMT_MTD.1	FMT_SMR.1	FPT_STM.1
5.1.1	FAU_GEN.1															X
5.1.2	FAU_GEN.2	X									X					
5.1.3	FAU_SAR.1	X														
5.1.4	FAU_SAR.2		X													
5.1.5	FAU_SAR.3		X													
5.1.6	FAU_SEL.1	X												X		
5.1.7	FAU_STG.1	X														
5.1.8	FAU_STG.3			X												
5.1.9	FAU_STG.4	X														
5.2.1	FAU_ACC.1					X										
5.2.2	FAU_ACF.1				X											
5.2.3	FDP_RIP.2															
5.2.4	Note 1															
5.3.1	FIA_ATD.1															
5.3.2	FIA_SOS.1															
5.3.3	FIA_UAU.1										X					
5.3.4	FIA_UAU.7									X						
5.3.5	FIA_UID.1															
5.3.6	FIA_USB.1								X							
5.4.1	FMT_MSA.1				O		O								X	
5.4.2	FMT_MSA.3											X			X	
5.4.3	FMT_MTD.1														X	
5.4.4	FMT_MTD.1														X	
5.4.5	FMT_MTD.1														X	

Section	CC Identifier	FAU_GEN.1	FAU_SAR.1	FAU_STG.1	FDP_ACC.1	FDP_ACF.1	FDP_IFC.1	FDP_IFF.1	FIA_ATD.1	FIA_UAU.1	FIA_UID.1	FMT_MSA.1	FMT_MSA.3	FMT_MTD.1	FMT_SMR.1	FPT_STM.1
5.4.6	FMT_MTF.1														X	
5.4.7	FMT_REV.1														X	
5.4.8	FMT_REV.1														X	
5.4.9	FMT_SMR.1										X					
5.5.1	FPT_AMT.1															
5.5.2	FPT_RVM.1															
5.5.3	FPT_SEP.1															
5.5.4	FPT_STM.1															

7.4 Rationale for Assurance Rating

This protection profile has been developed for a generalized environment with a moderate level of risk to the assets. It is intended that products used in these environments will be generally available, without modification to meet the security needs of the environment. As such it was determined the Evaluation Assurance Level 3 was the most appropriate.

7.5 Rational for SOF Rating

The strength of function rating of SOF-medium is consistent with the SFR FIA_SOS.1 by providing a ‘one off’ probability of guessing the password in 1,000,000. This SFR is in turn consistent with the security objectives described in section 7.2.

8.0 Notes on Deviations

This section contains notes on places where this protection profile deviated from version 2 of the Common Criteria. Each of these notes has been submitted as Common Criteria Request for Interpretations (CCRI) for inclusion in future versions of the Common Criteria.

Note 1 The CC's FDP_RIP components only specify resources being allocated to objects and does not address resources used directly by subjects, such as memory or registers. This explicit requirement was added to ensure coverage of these resources. The words are identical to FDP_RIP.2 except "subject" replaces "object".

Note 2 The CC's FIA_USB component used the term "appropriate security attributes", which is really too vague. The word "appropriate" was replaced with the word "following" and an assignment list was added. This allows the PP/ST to specify what attributes are needed to enforce the TSP. In addition, elements were added to cover any rules that were required to be enforced on attribute binding or changes.

Note 3 The word "take" was removed from FAU_STG.3 in order to make a grammatically correct sentence.

Note 4 The format of using sub-elements which appeared in the CC's FAU_GEN.1 was difficult to represent all the information in a clear fashion. The sub-elements were replaced by the use of a table, as the wording of the element adjusted to refer to the table, rather than the sub-elements.

Note 5 The CC's FAU_STG.4 did not provide a selection which only required that this capability be available. The words "be able to" were added in order to allow the option to make this a configurable feature.