TOC

Consistency Instruction Manual

For development of

US Government Protection Profiles (PP)

For use in

Basic Robustness Environments

Release 2.0

1 March 2004

Forward

(Back to TOC)

This Protection Profile (PP) Consistency Instruction Manual for Basic Robustness Environment was developed by the Protection Profile Review Board (PPRB) to identify and set forth a framework of consistent security requirements for the specification of products in environments requiring Basic robustness, based on Version 2.1 of the Common Criteria, International Standard 15408. Details of the complete Common Criteria may be found at http://csrc.nist.gov/cc

It is the intent of the PPRB that this manual be periodically updated. Feedback on its content may be forwarded to Ms. Jean Schaffer at jhschaf@missi.ncsc.mil.

If you are viewing this document online, you should activate your web toolbar (View, Toolbars, Web) to maximize the use of hyperlinks embedded throughout the document.

Record of Release

1. Preliminary Release 1.0, dated September 2002
2. Release 2.0 dated 1 March 2004

Table of Contents

I. Introduction

(Back to TOC)

NSA has produced a number of Common Criteria Protection Profiles in response to the Office of the Secretary of Defense (OSD) request for procurement guidance on IA technologies. This work is being performed to support new Department of Defense IA system policies (i.e., DoDD 8500.1 and DODI 8500.2). In November 2001, NSA and the National Institute of Standards and Technology agreed to work together to create a joint set of profiles that would represent the two organizations' collective interests.

With many profiles being developed by numerous organizations within NIST and NSA, it has become apparent that in order for the organizations to lead in this area, IA Protection Profile efforts need to be closely coordinated to facilitate representing a consistent strategic view to our customer base. Such consistency is important to create and maintain our customer's confidence in our products and guidance.

To this end, a corporate PP consistency-working group, called the PP Review Board (PPRB), has been formed to review all proposed PPs and work with the PP authors to offer comments to make them as consistent as possible. The first activity of this group was to review a number of Protection Profiles and offer comments to the authors on areas that should be addressed to improve consistency. In the context of this first review, a number of consistent items for Basic Robustness Profiles have been captured and recorded in this document that will offer Basic PP authors guidance on how to make U.S. Government PPs more consistent.

The document presents instructions for a PP author. The instructions are presented for all PP authors to consider and either **include** the recommendation in their PP **or justify** why the recommendation does not apply to the profile. This methodology will ensure that all PP authors address the minimal security considerations or perform an analysis as to why they are not addressed. Each instruction is self-contained and offers either text for specific sections of a PP or specific common criteria functional/security requirements so that all PP are consistent in addressing minimum-security concerns for Basic Robustness PP.

It should be noted that the final authority for the content of the PP is the PP owner. However, the profile must be consistent with other profiles of the same robustness thus the author should review other profile at the same robustness level. The author should also ensure that the functional requirements are consistent with the technology and may want to consult with other experts in the technology area.

As PP reviews continue, this guidance will be updated to offer new instructions as they become available.

II. Basic Robustness Definition

(BACK TO TOC)

Instruction 1: Characterize Robustness Level

(Back to TOC)

All PPs should contain a discussion characterizing the level of robustness TOEs compliant with the PP can achieve, thus allowing a user of the PP to determine if a compliant TOE is appropriate for the environment in which they intend to use the TOE. The PPRB created a discussion (included below) that provides a definition of factors for TOE environments as well as an explanation of how a given level of robustness is categorized.

The intent of these new sections is to have system integrator and product vendors clearly understand the concept of robustness, what products or systems designed to meet a specific robustness level are useful for, and the suitability of a level of robustness for their application.

DODI 8500.2 February 6, 2003 says, "Robustness describes the strength of mechanism (e.g., the strength of a cryptographic algorithm) and assurance properties (i.e., confidence measures taken to ensure proper mechanism implementation) for an IA solution. The more robust a particular component is, the greater the level of confidence in the protection provided to the security services it supports. The three levels of robustness are discussed in detail in Chapter 4 in the Information Assurance Technical Framework (IATF), reference (k). It is also possible to use non-technical measures to achieve the equivalent of a level of robustness. For example, physical isolation and protection of a network can be used to provide confidentiality. In these cases, the technical solution requirement may be reduced or eliminated."

Text:
Below is text (blue text) for inclusion as Appendix D of the Basic Robustness Protection Profile.

General Environmental Characterization

In trying to specify the environments in which TOEs with various levels of robustness are appropriate, it is useful to first discuss the two defining factors that characterize that environment: **value of the resources** and **authorization of the entities** to those resources.

In general terms, the environment for a TOE can be characterized by the authorization (or lack of authorization) the least trustworthy entity has with respect to the highest value of TOE resources (i.e. the TOE itself and all of the data processed by the TOE).

Note that there are an infinite number of combinations of entity authorization and value of resources; this conceptually "makes sense" because there are an infinite number of

potential environments, depending on how the resources are valued by the organization, and the variety of authorizations the organization defines for the associated entities. In the next section 1.2.2, these two environmental factors will be related to the robustness required for selection of an appropriate TOE.

VALUE OF RESOURCES

Value of the resources associated with the TOE includes the data being processed or used by the TOE, as well as the TOE itself (for example, a real-time control processor). "Value" is assigned by the using organization. For example, in the DoD low-value data might be equivalent to data marked "FOUO", while high-value data may be those classified Top Secret. In a commercial enterprise, low-value data might be the internal organizational structure as captured in the corporate on-line phone book, while high-value data might be corporate research results for the next generation product. Note that when considering the value of the data one must also consider the value of data or resources that are accessible through exploitation of the TOE. For example, a firewall may have "low value" data itself, but it might protect an enclave with high value data. If the firewall was being depended upon to protect the high value data, then it must be treated as a high-value-data TOE.

AUTHORIZATION OF ENTITIES

Authorization that entities (users, administrators, other IT systems) have with respect to the TOE (and thus the resources of that TOE, including the TOE itself) is an abstract concept reflecting a combination of the trustworthiness of an entity and the access and privileges granted to that entity with respect to the resources of the TOE. For instance, entities that have total authorization to all data on the TOE are at one end of this spectrum; these entities may have privileges that allow them to read, write, and modify anything on the TOE, including all TSF data. Entities at the other end of the spectrum are those that are authorized to few or no TOE resources. For example, in the case of a router, non-administrative entities may have their packets routed by the TOE, but that is the extent of their authorization to the TOE's resources. In the case of an OS, an entity may not be allowed to log on to the TOE at all (that is, they are not valid users listed in the OS's user database)**.**

It is important to note that authorization **does not** refer to the **access** that the entities actually have to the TOE or its data. For example, suppose the owner of the system determines that no one other than employees was authorized to certain data on a TOE, yet they connect the TOE to the Internet. There are millions of entities that are not **authorized** to the data (because they are not employees), but they actually have connectivity to the TOE through the Internet and thus can attempt to access the TOE and its associated resources.

Entities are characterized according to the value of resources to which they are authorized; the extent of their authorization is implicitly a measure of how trustworthy the entity is with respect to compromise of the data (that is, compromise of any of the applicable security policies; e.g., confidentiality, integrity, availability). In other words,

in this model the greater the extent of an entity's authorization, the more trustworthy (with respect to applicable policies) that entity is.

SELECTION OF APPROPRIATE ROBUSTNESS LEVELS

Robustness is a characteristic of a TOE defining how well it can protect itself and its resources; a more robust TOE is better able to protect itself. This section relates the defining factors of IT environments, authorization, and value of resources to the selection of appropriate robustness levels.

When assessing any environment with respect to Information Assurance the critical point to consider is the likelihood of an attempted security policy compromise, which was characterized in the previous section in terms of entity authorization and resource value. As previously mentioned, robustness is a characteristic of a TOE that reflects the extent to which a TOE can protect itself and its resources. It follows that as the likelihood of an attempted resource compromise increases, the robustness of an appropriate TOE should also increase.

It is critical to note that several combinations of the environmental factors will result in environments in which the likelihood of an attempted security policy compromise is similar. Consider the following two cases:

The first case is a TOE that processes only low-value data. Although the organization has stated that only its employees are authorized to log on to the system and access the data, the system is connected to the Internet to allow authorized employees to access the system from home. In this case, the least trusted entities would be unauthorized entities (e.g. non-employees) exposed to the TOE because of the Internet connectivity. However, since only low-value data are being processed, the likelihood that unauthorized entities would find it worth their while to attempt to compromise the data on the system is low and selection of a basic robustness TOE would be appropriate.

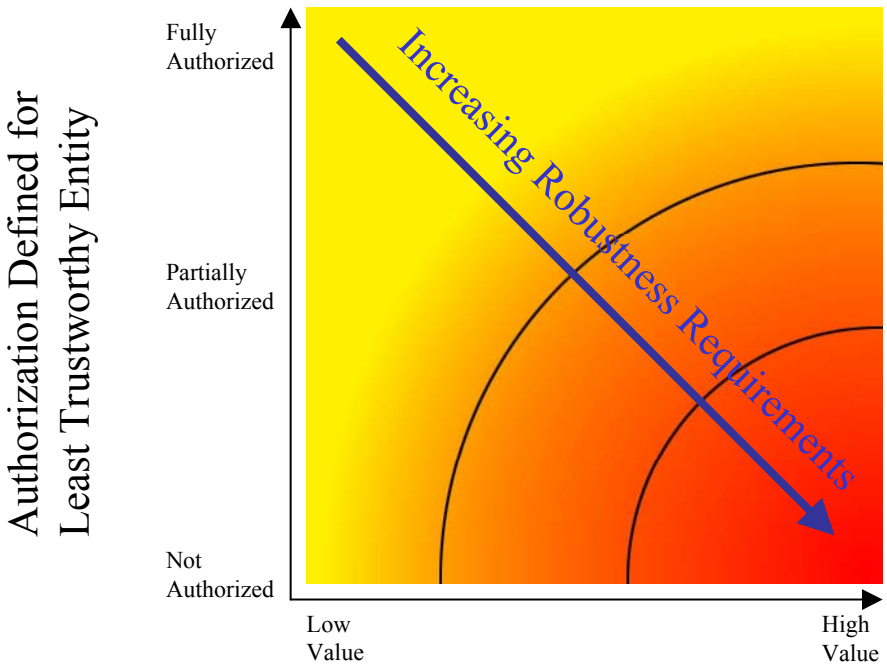
The second case is a TOE that processes high-value (e.g., classified) information. The organization requires that the TOE be stand-alone, and that every user with physical and logical access to the TOE undergo an investigation so that they are authorized to the highest value data on the TOE. Because of the extensive checks done during this investigation, the organization is assured that only highly trusted users are authorized to use the TOE. In this case, even though high value information is being processed, it is unlikely that a compromise of that data will be attempted because of the authorization and trustworthiness of the users and once again, selection of a basic robustness TOE would be appropriate.

The preceding examples demonstrated that it is possible for radically different combinations of entity authorization/resource values to result in a similar likelihood of an attempted compromise. As mentioned earlier, the robustness of a system is an indication of the protection being provided to counter compromise attempts. Therefore, a basic robustness system should be sufficient to counter compromise attempts where the likelihood of an attempted compromise is low. The following chart depicts the

"universe" of environments characterized by the two factors discussed in the previous section: on one axis is the authorization defined for the least trustworthy entity, and on the other axis is the highest value of resources associated with the TOE.

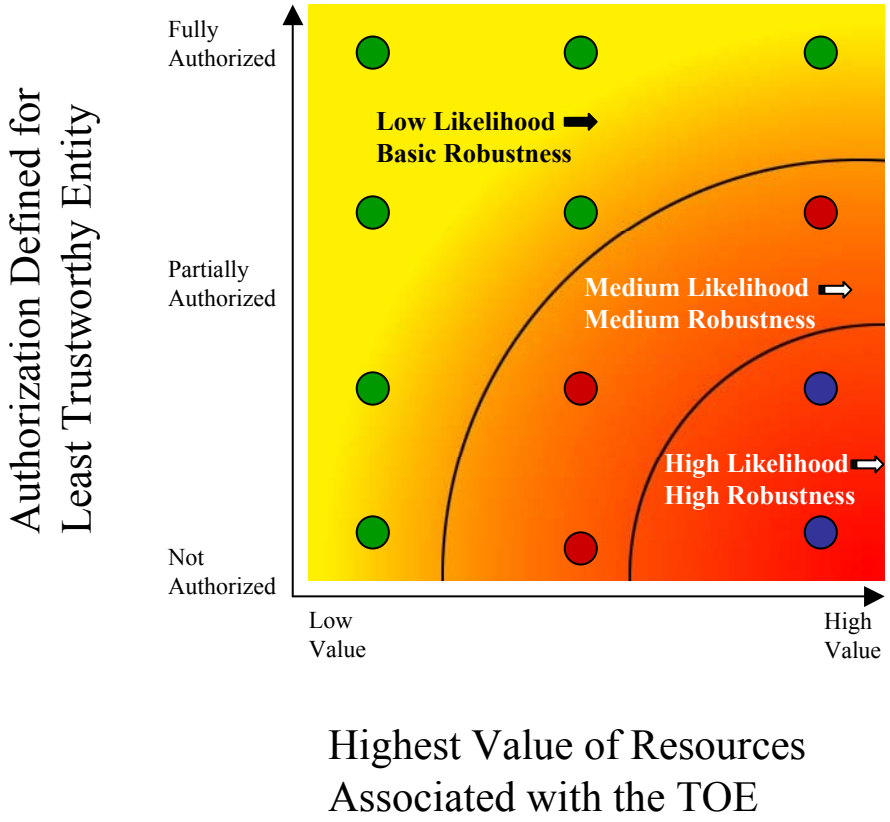
As depicted in the following figure, the robustness of the TOEs required in each environment steadily increases as one goes from the upper left of the chart to the lower right; this corresponds to the need to counter increasingly likely attack attempts by the least trustworthy entities in the environment. Note that the shading of the chart is intended to reflect- the notion that different environments engender similar levels of "likelihood of attempted compromise", signified by a similar color. Further, the delineations between such environments are not stark, but rather are finely grained and gradual.

While it would be possible to create many different "levels of robustness" at small intervals along the "Increasing Robustness Requirements" line to counter the increasing likelihood of attempted compromise due to those attacks, it would not be practical nor particularly useful. Instead, in order to implement the robustness strategy where there are only three robustness levels: Basic, Medium, and High, the graph is divided into three sections, with each section corresponding to a set of environments where the likelihood of attempted compromise is roughly similar. This is graphically depicted in the following chart.



In this second representation of environments and the robustness plane below, the "dots" represent given instantiations of environments; like-colored dots define environments with a similar likelihood of attempted compromise. Correspondingly, a TOE with a given robustness should provide sufficient protection for environments characterized by like-colored dots. In choosing the appropriateness of a given robustness level TOE PP for an environment, then, the user must first consider the lowest authorization for an entity as well as the highest value of the resources in that environment. This should result in a "point" in the chart above, corresponding to the likelihood that that entity will attempt to compromise the most valuable resource in the environment. The appropriate robustness level for the specified TOE to counter this likelihood can then be chosen.

The difficult part of this activity is differentiating the authorization of various entities, as well as determining the relative values of resources; (e.g., what constitutes "low value" data vs. "medium value" data). Because every organization will be different, a rigorous definition is not possible. In <PP Section>[1] of this PP, the targeted threat level for a Basic robustness TOE is



characterized. This information is provided to help organizations using this PP - ensure that the functional requirements specified by this Basic robustness PP are appropriate for their intended application of a compliant TOE.

[1] The PP author should insert the section of the PP that describes the TOE Environment.

Instruction 2: Accommodating Software Only TOEs

(Back to TOC)

Requiring hardware for Basic Robustness TOEs is not mandated by the PPRB, "Software only TOEs" can be accommodated for basic robustness. If the PP author feels that additional assurance is necessary, hardware may be included as part of the TOE.

Experience has shown that many security compromises occur when products are "composed"; that is, individual products that may be, by themselves, trustworthy, result in a vulnerable design when they are integrated together as a composite product. In order to provide the assurance necessary for products to be integrated into medium robustness environments, it is generally necessary to require that certain components of a product be evaluated as part of a TOE to give high confidence that the product is tamperproof and that the security policy is always invoked (as opposed to allowing an evaluation sponsor to exclude the component from the definition of the TOE and relegate it to the environment). A particular component of note for all medium robustness products is the product's hardware. Because it is important for medium robustness products to show, through analysis and testing of an evaluation, that they are truly tamperproof and always invoke the correct policy, the hardware associated with medium robustness product hardware should almost always, be specified as part of the TOE and demonstrated as compliant to a medium robustness PP.

However, basic robustness environments are defined as countering threats introduced by inadvertent errors or casually mischievous users. Given this, it is appropriate for the countermeasures and assurances required for products intended for this environment to not require significant security features or assurances: "**best commercial practice**" will be sufficient to protect information in a basic robustness environment. Therefore, the scope and rigor of a basic robustness PP should not be the same as that of a medium robustness PP. Specifically, it is acceptable for a basic robustness PPs to specify a "Software Only" TOE. This allows evaluation sponsors (e.g., vendors) to scope the evaluation work to components for which they can readily provide information while at the same time offering end users some statements about what the evaluated products offers in the context of security features.

To allow for a "Software Only" TOE in a PP, requires four actions for the PP author:

1. The PP author must make it clear in the TOE Description that PP compliant TOEs will be "Software only".
2. The PP author must change the O.SELF_PROTECTION objective to the O.PARTIAL_SELF_PROTECTION objective.
3. The PP author must *explicitly state* the domain isolation requirement (FPT_SEP.*).
4. The PP author must include FPT_STM.1 as a requirement on the IT Environment.

The TOE Description (Section 2 of a PP) describes the general nature of PP compliant TOEs and includes things such as a general description of the TOE, the "technology

type", the TOEs security features, and other salient attributes that characterize what it is that the PP author is requesting to be analyzed and tested. Whether a TOE is expected to comprise software and hardware or just software is part of this description.

Given the nature of a PP compliant TOE is described in the TOE Description, the objectives and functional requirements must ultimately reflect this description. Software Only TOE properties are instantiated in Section 3 of the PP (i.e., the Functional Requirements section) by creating explicitly stated requirements in place of FPT_SEP.*. The need for explicitly stated requirements is that when invoked, the current FPT_SEP.* Common Criteria Requirement requires the TOE (not its environment) to protect itself from external interference and tampering. Typically, "Software Only" technology cannot fully meet these requirements as written. Software Only TOEs should be expected to work in the context of their hardware environment to aid in enforcing domain separation but cannot be required to fully counter the threats without hardware. Therefore, PP authors should use the *explicitly stated* requirements for domain separation when attempting to accommodate "Software Only" TOEs.

In addition, PP authors should also include an IT Environmental requirement in Section 3 of the PP (i.e., the Environmental section) that describes the domain isolation requirements that the IT Environment must meet. An example of such a requirement is:

> FPT_SEP_ENV.1 The TSF Environment shall provide hardware that provides virtual memory management and at least two execution rings for executing software.

Finally, FPT_STM.1 requires hardware; even if the TOE is depending on an NTP server on the network. For this reason, software-only TOEs must put FPT_STM.1 in the IT Environment. If the TOE software is responsible for any part of the FPT_STM.1 requirement (e.g., implements an NTP client) then FPT_STM.1 should also be placed on the TOE.

Suggested Text for O.PARTIAL_SELF_PROTECTION

> O.PARTIAL_SELF_PROTECTION: The TSF will maintain a domain for its own execution that protects itself and its resources from external interference, tampering or unauthorized disclosure, through its own interfaces.

Suggested Text for FPT_SEP

Below is the suggested text for a Common Criteria *explicitly stated* requirement for FPT_SEP.1

> FPT_SEP_(EXP).1 The TSF shall maintain a security domain that protects it from interference and tampering by untrusted subjects initiating actions through its own TSFI.

FPT_SEP_(EXP).2 The TSF shall enforce separation between the security domains of subjects in the TOE Scope of Control.

Instruction 3: Uses of Basic Robustness

(Back to TOC)

The PPRB recognized the importance of a clear understanding of the TSE specified in terms of applicable assumptions, threats and policies which are related to or appropriate for a particular robustness levels.

Therefore, it is suggested that PP authors include in section 3 of all PPs a discussion relating the specified TOE robustness level to the formation of applicable assumptions, threats and policies of the TOE security environment (TSE).

Suggested Text for Basic Robustness PPs:

Basic robustness TOEs falls in the upper left area of the previously discussed robustness figures. A Basic Robustness TOE is considered sufficient for low threat environments or where compromise of protected information will not have a significant impact on mission objectives. This implies that the motivation of the threat agents will be low in environments that are suitable for TOEs of this robustness. In general, basic robustness results in "good commercial practices" that counter threats based in casual and accidental disclosure or compromise of data protected by the TOE.

Threat agent motivation can be considered in a variety of ways. One possibility is that the value of the data process or protected by the TOE will generally be seen as of little value to the adversary (i.e., compromise will have little or no impact on mission objectives). Another possibility, (where higher value data is processed or protected by the TOE) is that procuring organizations will provide other controls or safeguards (i.e., controls that the TOE itself does not enforce) in the fielded system in order to increase the threat agent motivation level for compromise beyond a level of what is considered reasonable or expected to be applied.

INSTRUCTION 4: ASSURANCE REQUIREMENTS FOR BASIC ROBUSTNESS

(Back to TOC)

The agreed upon Security Assurance Requirements drawn from the Common Criteria for Information Technology Security Evaluation, Part 3, dated Aug. 99, Version 2.1 of CCIB-99-031 which collectively define "Basic Robustness" include the following:

All of the assurance requirements included in Evaluated Assurance Level (EAL) 2 augmented with the following additions:

- ALC_FLR.2, Flaw Remediation
- AVA_MSU.1 Examination of Guidance

The following is a list of the assurance requirements needed for Basic Robustness:

Family	Assurance Components	Assurance Components Description
Configuration Management	ACM_CAP.2	Configurations items
Delivery and Operation	ADO_DEL.1	Delivery procedures
	ADO_IGS.1	Installation, generation, and start-up procedures
Development	ADV_FSP.1	Informal Functional specification
	ADV_HLD.1	Descriptive high-level design
	ADV_RCR.1	Informal correspondence demonstration
Guidance Documents	AGD_ADM.1	Administrator guidance
	AGD_USR.1	User guidance
Life Cycle Support	**ALC_FLR.2**	**Flaw Reporting Procedures**
Tests	ATE_COV.1	Evidence of coverage
	ATE_FUN.1	Functional testing
	ATE_IND.2	Independent testing - sample
Vulnerability Assessment	**AVA_MSU.1**	**Examination of guidance**
	AVA_SOF.1	Strength of TOE security function evaluation
	AVA_VLA.1	Developer Vulnerability analysis

III. GENERAL INFORMATION INSTRUCTIONS

(Back to TOC)

INSTRUCTION 5: CONTENT AND OUTLINE OF A PROTECTION PROFILE

(Back to TOC)

Title page

The Title Page will include the title, version and date of the protection profile. See Instruction 6 and Appendix D for details about the title page

1. Introduction to the Protection Profile

1.1 PP Identification
1.2 PP Overview of the protection profile
1.2.1 General Environmental Characterization
1.3 Conventions – See instruction 13
1.4 Glossary of terms – See instruction 14
1.5 Document Organization

2. TOE Description

2.1 Product type
2.2 Toe Definition
2.3 General TOE functionality
2.4 TOE Operational environment

3. Security Environment

3.1 Threats – See instruction 8
3.2 Organizational Security Policies – See instruction 9
3.3 Assumptions – See instruction 7

4. Security Objectives

4.1 TOE Security Objectives – See instruction 9
4.2 Environment Security Objectives - See instruction 10

5. IT Security Requirements

5.1 TOE Security Functional Requirements – See instructions 15-25
5.2 Security Requirements for the IT Environment - See instruction 10
5.3 TOE Security Assurance Requirements – See instruction 4

6. Rationale

6.1 Rationale foe TOE Security Objectives - See Appendix A
6.2 Rationale for the security objectives and security functional requirements for the environment
6.3 Rationale for TOE Security Requirements - See Appendix B
6.4 Rationale for assurance requirements
6.5 Rational for strength of function claim
6.6 Rational for satisfying all dependencies
6.7 Rational for explicit requirements
5.8 Rationale for not addressing consistency instructions

7. Appendices:

A. References
B. Glossary - See instruction 14
C. Acronyms
D. Robustness Environment Characterization – See instruction 1

INSTRUCTION 6: FORMAT FOR THE TITLE PAGE OF A PROTECTION PROFILE

(Back to TOC)

In general, whole numbers (starting with 1) will be reserved for NIAP validated profiles, and decimal numbers (starting with 0.1) will be used for draft profiles, which are released for review outside of the immediate development team. The team may use finer granularity for its internal coordination and tracking purposes

NIAP Validated profile will be whole numbers starting with 1 and increased by 1 for each new revision that get NIAP validated. Examples will be “Version 1”, “Version 3”, etc not “Version 1.0” or “Version 3.0.”

Draft profiles will start decimal numbers starting with 0.1 and increased by .1 for each new draft released outside of the development team. Examples will be “Version 0.1”, or “Version 0.3”. Drafts are documents that have been written and are under going various stages of review. Once a draft is written and released for the first review, it will be labeled “Version 0.1”. If no changes are required during a review the version number will remain the same, however if it is determined that changes are required the draft version number will be increase by .1 indicating the changes were made and the review process continues (even if it is back to the same review step).

When it is required to update a NIAP validated Protection Profile, the updated drafts will be numbered “Version 1.1”, or “Version 1.2”, etc. Once the NIAP validates the new draft, it will get a new NIAP validated whole number 2, 3, etc.

In addition to the version number, the profile will contain a title of the profile and the date of the proposed version. The format of the date will be yyyymmdd. The title of the document should be provided in the following format "U.S. Government Protection Profile for (technology) used in (Robustness Level) Environments." Since we are now in a joint NSA/NIST process all profile will be U.S. Government and not DoD specific.

See appendix D for the template that shall be used by the Profile Author. The author shall fill in the technology area, date, version number and use cover sheet for their Profile.

INSTRUCTION 7: ASSUMPTIONS

(Back to TOC)

Assumptions (included in Section 3 of the PP) are defined as ***non-IT*** items that the TOE itself cannot implement or enforce. Assumptions should not be used to specify functional requirements on the IT environment; that should be done with a threat or policy statement. For instance, a valid assumption might be "All administrators will be trained in the secure administration of the TOE." The TOE has no control over whether the administrators are trained or not, so this is a valid assumption. An invalid assumption might be "All users are authenticated before taking any action on the TOE." Since the TOE (or IT environment) could implement this, it is not a valid assumption.

In addition, it is useful to readers of the PP to list assumptions necessary for the TOE to work correctly.

From the initial review of several PP, the PPRB identified a few assumptions that seem to be frequently specified by PP authors. The text below proposes consistent names and descriptions for these commonly included assumptions. Note that not all assumptions will be valid for all PPs. PP authors need to determine if whether specific assumptions apply to the TOE being described in the PP.

	Text
A.NO_GENERAL_PURPOSE[2]	The administrator ensures there are no general-purpose computing or storage repository capabilities (e.g., compilers, editors, or user applications) available on the TOE.
A.NO_EVIL	Administrators are non-hostile, appropriately trained and follow all administrator guidance.
A.PHYSICAL	It is assumed that the IT environment provides the TOE with appropriate physical security, commensurate with the value of the IT assets protected by the TOE.

[2] This assumption should be used only on "server"-type TOEs that should have no general-purpose functionality available to untrusted users. It makes sense, for example, for a firewall or a router, but does not make sense for an operating system or someone's desktop computer.

INSTRUCTION 8: DESCRIBING THREATS

(Back to TOC)

Threats (included in Section 3 of the PP) are stated as risks to security that the TOE will mitigate or eliminate. Therefore, threat statements must **not** include situations in which the TOE plays no part (i.e., those that are completely addressed by the environment), threats the TOE cannot recognize (e.g., the TOE may be incorrectly configured), or threats to the TOE itself that would not exist without the TOE (e.g., the TOE may contain Trojan horses).

The PPRB recognized the importance of a clear understanding of the basis for specifying appropriate threats for a given robustness level and therefore, requires the inclusion in section 3 of all PPs, a discussion that will establish the context of how to formulate applicable threats for a given robustness level. The following text should be included in all PPs to explain to PP authors and reviewers, how the itemized threats as described in the TSE section were formulated.

Text for Describing the Threat Environment

Threat Agent Characterization

In addition to helping define the robustness appropriate for a given environment, the threat agent is a key component of the formal threat statements in the PP. Threat agents are typically characterized by a number of factors such as *expertise*, *available resources*, and *motivation*. Because each robustness level is associated with a variety of environments, there are corresponding varieties of specific threat agents (that is, the threat agents will have different combinations of motivation, expertise, and available resources) that are valid for a given level of robustness. The following discussion explores the impact of each of the threat agent factors on the ability of the TOE to protect itself (that is, the robustness required of the TOE).

The *motivation* of the threat agent seems to be the primary factor of the three characteristics of threat agents outlined above. Given the same expertise and set of resources, an attacker with low motivation may not be as likely to attempt to compromise the TOE. For example, an entity with no authorization to low value data none-the-less has low motivation to compromise the data; thus a basic robustness TOE should offer sufficient protection. Likewise, the fully authorized user with access to highly valued data similarly has low motivation to attempt to compromise the data, thus again a basic robustness TOE should be sufficient.

Unlike the motivation factor, however, the same can't be said for *expertise*. A threat agent with low motivation and low expertise is just as unlikely to attempt to compromise a TOE as an attacker with low motivation and high expertise; this is because the attacker with high expertise does not have the motivation to compromise the TOE even though they may have the expertise to do so. The same argument can be made for *resources* as well.

Therefore, when assessing the robustness needed for a TOE, the motivation of threat agents should be considered a "high water mark". *That is, the robustness of the TOE should increase as the motivation of the threat agents increases.*

Having said that, the relationship between expertise and resources is somewhat more complicated. In general, if resources include factors other than just raw processing power (money, for example), then expertise should be considered to be at the same "level" (low, medium, high, for example) as the resources because money can be used to purchase expertise. Expertise in some ways is different, because expertise in and of itself does not automatically procure resources. However, it may be plausible that someone with high expertise can procure the requisite amount of resources by virtue of that expertise (for example, hacking into a bank to obtain money in order to obtain other resources).
It may not make sense to distinguish between these two factors; in general, it appears that the only effect these may have is to lower the robustness requirements. For instance, suppose an organization determines that, because of the value of the resources processed by the TOE and the trustworthiness of the entities that can access the TOE, the motivation of those entities would be "medium". This normally indicates that a medium robustness TOE would be required because the likelihood that those entities would attempt to compromise the TOE to get at those resources is in the "medium" range. However, now suppose the organization determines that the entities (threat agents) that are the least trustworthy have no resources and are unsophisticated. In this case, even though those threat agents have medium motivation, the likelihood that they would be able to mount a successful attack on the TOE would be low, and so a basic robustness TOE may be sufficient to counter that threat.

It should be clear from this discussion that there is no "cookbook" or mathematical answer to the question of how to specify exactly the level of motivation, the amount of resources, and the degree of expertise for a threat agent so that the robustness level of TOEs facing those threat agents can be rigorously determined. However, an organization can look at combinations of these factors and obtain a good understanding of the likelihood of a successful attack being attempted against the TOE. Each organization wishing to procure a TOE must look at the threat factors applicable to their environment; discuss the issues raised in the previous paragraph; consult with appropriate accreditation authorities for input; and document their decision regarding likely threat agents in their environment.

The important general points we can make are:

- The motivation for the threat agent defines the upper bound with respect to the level of robustness required for the TOE
- A threat agent's expertise and/or resources that is "lower" than the threat agent's motivation (e.g., a threat agent with high motivation but little expertise and few resources) may lessen the robustness requirements for the TOE (see next point, however).
- The availability of attacks associated with high expertise and/or high availability of resources (for example, via the Internet or "hacker chat rooms") introduces a

problem when trying to define the expertise of, or resources available to, a threat agent.

INSTRUCTION 9: THREATS, POLICIES, OBJECTIVES AND REQUIREMENTS

(Back to TOC)

Basic Robustness PPs should contain relevant **threats, policies and associated objectives and requirements** for the Basic Robustness level, and use a consistent naming convention and description. The PPRB has formulated a list of **threats, policies, and objectives** that must be considered for all Basic Robustness TOEs, and a methodology for instantiating these in a PP. Each threat or policy has one or more objectives that address the stated threat or policy, and each objective in turn has requirement components associated with it that address the stated objective and mitigate or implement the threat or policy.

Unfortunately, cutting-and-pasting of all of these items without careful consideration is not appropriate. Reasons include:

- a threat may not apply to a technology;
- a threat or policy may be applicable but may need to be tailored in a technology-specific way; or
- although the threat may be applicable for the technology, the way in which it is countered, or the resources to which it applies, may be different depending on the technology. This might necessitate a change in the objective and/or requirement components; or
- some technologies may have threats that are not provided in this guidance that need to be countered, or policies that need to be met. For these additional threats or policies, additional objectives may need to be formulated, and requirements added.

Additionally, for most threat/objective/requirement mappings the rationale (how a set of objectives satisfies a threat or policy, and how a set of requirement components meets an objective) will have to be written "from scratch" to reflect the unique aspects of the technology. Some rational is included in this document for reference and possible use in Basic robustness PPs. Care should be taken to review it to ensure its validity before it is included.

PP Creation Methodology Overview

In order to enhance consistency in writing PPs, the PPRB has formulated a methodology that can be used by PP authors in creating a substantial portion of the PP. There are several things to note about this methodology:

- This methodology has been used to produce quality PPs that are consistent with the PPRB guidance given in Table 7, *Applicable Threats, Policies, Objectives and Requirements for Basic Robustness TOEs*. **This does not mean that other methodologies cannot be used.** If the PP authors have a different approach that will

yield a PP that is consistent with the PPRB guidance, they are welcome to use it.

- While the PP writing team may not use the methodology described below, they should still use the threats, objectives, and requirements listed in Table 7 to ensure consistency with other Basic Robustness TOEs.

- The following methodology is for the creation of significant parts of the PP. However, additional work will have to be done by the PP writing team to complete the document.

It is critical in writing a PP that the requirements support the objectives and either mitigate the threats, or implement the policies stated in the PP. The CC framework calls for this to be documented in "*rationale*" sections: one detailing how the objectives (and associated requirements) mitigate a threat or implement a policy, and one detailing how the requirements implement the objectives (see Instruction 12 for more information on writing the Rationale sections). It is important to note that because the threat/policy to objective rationale section has to detail *how the applicable requirements from the objective mitigate the threat (or implement the policy)*, it is important for the PP authors to "keep track" of how the threats/polices map to objectives, and what requirements from those objectives relate to the threat/policies.

The PPRB has found that using a spreadsheet to keep track of this information is helpful. Although such a spreadsheet *is not* part of the PP itself, it can be a useful tool for PP authors in tracking the association between threats/policies, objectives, and requirements. In Appendix C of this guidance a spreadsheet has been prepared that has been "pre-loaded" with the information in Table 7. The PP authors can update this spreadsheet as they are working through the steps in the methodology so that when they are ready to write the rationale sections, they can ensure that they have accurately captured the relationship between all three "levels" in the requirements decomposition (those three levels being: threats/policies, objectives, and requirements).

Using Table 7: Applicable Threats, Policies, Objectives and Requirements for Basic Robustness TOEs

Table 7 consists of three columns. The first column indicates the threats and policies that the PP author must include in their Basic Robustness PP. Each of the threats is mitigated by one or more objectives; likewise, each of the policies is implement by one or more objectives. For each threat/policy, the objective or objectives that mitigate/implement it are listed in the second column. Note that the same objective may be listed more than once in this second column, depending on how many of the threats/policies it applies to.

Each objective is implemented by one or more requirements ("components" in CC terminology). While multiple requirement components may be used to implement an entire objective, in some cases only a subset of those requirement components are used to counter a specific threat or implement a specific policy. This is reflected in the table by listing in column 3 only those requirements that apply to the particular threat or policy in column 1.

For instance, from Table 7 the PPRB suggests that O.TOE_ACCESS be implemented by FIA_AFL.1-NIAP-0425, FIA_ATD.1, FIA_UID, FIA_UAU, FTA_SSL.1, FTA_SSL.2, FTA_SSL.3, and AVA_SOF. O.TOE_ACCESS partially mitigates the T.MASQUERADE threat, fully mitigates the T.UNATTENDED_SESSION threat, and partially implements the P.ACCOUNTABILITY policy. However, not all of the requirements associated with O.TOE_ACCESS are applicable to all of the threats and policies that O.TOE_ACCESS is associated with (e.g., only the FIA_UID component of O.TOE_ACCESS is used to implement P.ACCOUNTABILITY). This is why there may be different sets of requirements listed in column 3 for the same objective.

The last column of Table 7 contains notes on the information in that row. It may draw attention to the threat/policy, the objective, or the requirement. Where the PPRB is recommending specific text (e.g., an assignment, selection or refinement) be used for a requirement, it may refer the PP authors to another instruction that contains the text the PP authors should use.

The PPRB suggests that the PP authors make a "working copy" of Table 7 so that if threats/policies are added, objectives added or changed, or when requirements are added or tailored, a centralized record can be maintained by modifying the copy of Table 7 appropriately. This will make it easier to create the CC-mandated tables that will appear in the PP in later steps in the methodology below. It is important to note that the only difference between this working copy of Table 7 and the Excel spreadsheet mentioned above and contained in Appendix C is that the Excel spreadsheet does not have the text associated with the threats and objectives, so that it can be more easily be viewed "all at once".

PP Creation Methodology

The suggested methodology for incorporating the information in Table 7 into a PP is described in the following steps. The overall approach is for the PP author to start at the beginning of Table 7 and address the first threat, then the objectives that apply to that threat, and finally the components from those objectives that mitigate the threat. The PP authors then address the next threat-objective-component "thread" until all threats and policies have been addressed.[3] After the PP authors ensure that the technology-specific details are covered, the PP material (various tables) is created and the rationale written. The details of this process is as follows:

1. The PP authors select the first (or next, for subsequent iterations) threat or policy provided in the Table 7. *Applicable Threats, Policies, Objectives and Requirements for Basic Robustness TOEs*. They should review the threat/policy statement to ensure its applicability to the subject PP. Most threats/policies will apply directly to the technology being specified in the PP; if there are technology-specific aspects to a threat, the PP authors should capture these aspects in the threat-to-objective rationale (see step 11) rather

[3] While it is certainly feasible to perform the activity by first doing all of the threats/policies, then doing all of the objectives, and then doing all of the requirement components, the methodology described above appears to reduce iteration on the part of the PP authors.

than try to create a new threat. Although a threat/policy may have to be tailored for a specific technology, this should be rare. Most threats/policies in Table 7 are sufficient so that no tailoring is necessary.

2. If the threat/policy is not applicable to the technology, a short justification will need to be included in Table 2 *Basic Robustness Threats **Not** Applicable to the TOE*. See Step 9 for placement of this table. It should be noted that placing a threat/policy from Table 7 into this category should be rare. The PP authors must be careful to distinguish threats that really don't apply because of the nature of the technology from threats that can't be countered because current instantiations of the technology do not include the required features.

3. If the threat/policy is applicable, then the objectives associated with the threat/policy in the table should be examined for validity. Note that the same objective may apply to multiple threats/policies, and thus may appear multiple times in the table (for example, O.RESIDUAL_INFORMATION is associated with T.AUDIT_COMPROMISE, T.RESIDUAL_DATA, and T.TSF_COMPROMISE). This means the PP authors will have to ensure that any text added or modified for an objective is applicable for all threats/policies to which that objective applies. In some cases, new objectives may need to be created; if so, the PP authors should ensure that the objective statements are consistent (with respect to format and level of detail) with those in the table.

4. Finally, the requirements components associated with each objective for the given threat/policy should be examined. The last column of Table 7 makes references to some instructions containing actual requirement component text (for example FAU_GEN.1 and instruction 15 of this document); the PPRB feels that this text should be included in the PP verbatim unless there is good justification for not doing so. Such text includes assignments, selections, etc. that is important to keep intact from a consistency perspective across all Basic Robustness PPs. In reviewing a Basic Robustness PP, the PPRB will note requirements that were not included verbatim, and will ask the PP authors for a rationale for omitting the recommended text. The PP authors should therefore ensure that when the decision is made to omit the recommend requirement text, a justification for this action is written and submitted with the PP for review by the PPRB.

   The PP authors should check to ensure that, for each requirement component chosen, the requirement component (1) applies to the objective and (2) mitigates some aspect of the threat/policy. The PP authors may want to make notes for the rationale section while they are doing this (see steps x and y, below). This step will be the most time consuming, and the PP authors may find they need to create new objectives, new threats/policies, etc. in the course of selecting components.

5. The PP authors then repeat steps 1 through 4 for each of the threats and policies listed in Table 7.

6. After the PP authors have gone through all of the threats and policies in the table, they need to consider if there are any technology-specific threats that need to be met by compliant TOEs. When considering such threats, the PP authors should consider whether the threat is appropriate for the Basic Robustness environment and whether the threat may be covered by an existing threat or policy. If the PP authors identify technology-specific aspects of an existing Basic Robustness threat, the PP authors should ensure that those aspects are captured in the threat-to-objective rationale statement (see step 11) as opposed to creating *new* technology-specific threats. For each new threat that is created, the objectives that will counter that threat should be either picked from existing objectives or (more likely) created by the PP authors, and components picked that meet the objective and mitigate the threat. The policies identified in Table 7 should be sufficient for all Basic Robustness TOEs. It is generally not necessary to create additional technology-specific policies because the requirements that would be derived from such policies would already be covered by existing threats and policies.

7. After performing the above steps, the PP authors should review the components to ensure that all desired functionality is included. If it is determined that some desired functionality is omitted, the PP authors should review the threat and policy statements to determine if the functionality is needed to counter one of the existing threats or implement one of the existing policies. In the unlikely event that no applicable threat or policy is found, the PP authors should devise a threat or policy statement (and associated objective) to which the functionality would apply, and then choose the appropriate components from the CC to require the functionality.

At the completion of step 7, all of the threats, policies, objectives, and requirements for the technology should be identified. If the PP authors have been modifying the working copy of Table 7 with updates to the threats, policies, objectives, and requirement component identifiers, the modified table will aid the team in their next tasks: creation of the threat, policy, and objective tables, and creation of the rationale.

8. The PP writing team should next construct a threat table (like Table 1 below) for the TOE Environment section of the PP that details all of the threats that apply to the TOE. The table should consist of each threat label, followed by the threat text. The threats should be in alphabetical order. A sample format follows:

Table 1 Basic Robustness Applicable Threats

T.AUDIT_COMPROMISE	A user or process may view audit records, cause audit records to be lost or modified, or prevent future audit records from being recorded, thus masking a user's action.

A similarly formatted table should be created for the policies and included in

the TOE environment section.

9. For those threats found to be not applicable to the TOE because the threat does not "make sense" for the technology area (see step 2 above), the PP authors should construct a table such as Table 2 below that details the threat label, the text of the threat, and a short rationale detailing why the threat is not applicable for the technology.

Table 2 Basic Robustness Threats NOT Applicable to the TOE

Threat Name	Threat Definition	Rationale for NOT Including this Threat
T.ADMIN_ERROR	An administrator may incorrectly install or configure the TOE resulting in ineffective security mechanisms.	There are no administrators on compliant TOEs.

10. The PP writing team should then construct an table of objectives for the TOE Objectives section of the PP that details all of the objectives. The objectives should be drawn from two sources. First, for each assumption on the IT environment (see Instruction 10) an objective for the IT environment should be created (see Table 3). Additionally, if a threat is mitigated (or a policy implemented) by both the TOE and the IT Environment, then an objective for the environment (in addition to the objective(s) for the TOE listed in Table 7) should be created for each of these. The environmental objectives should have a tag of "OE.*assumption_tag*", where *assumption_tag* is the tag associated with the assumption. For example, for the assumptions given in Instruction 7:

Table 3 Objectives for the IT Environment

OE. NO_GENERAL_ PURPOSE	There will be no general-purpose computing or storage repository capabilities (e.g., compilers, editors, or user applications) available on the TOE.
OE.NO_EVIL	Sites using the TOE shall ensure that administrators are non-hostile, appropriately trained and follow all administrator guidance.
OE.PHYSICAL	Physical security will be provided within the domain for the value of the IT assets protected by the operating system and the value of the stored, processed, and transmitted information.

Second, all objectives generated in steps 1 through 6 need to be captured in an objective table (in alphabetical order). The format is similar and is shown in Table 3:

Table 4 TOE Objectives

O.ADMIN_GUIDANCE	The TOE will provide administrators with the necessary information for secure management.
O.AUDIT_GENERATION	The TOE will provide the capability to detect and create records of security-relevant events associated with users.

11. The threat/policy-objective rationale section should be created next. In writing this rationale, the PP authors should use the format shown in Table 5.

Table 5 Threat/Policy to Objective Rationale Table

Threat/Policy	Objectives Addressing the Threat	Rationale
T.ADMIN_ERROR An administrator may incorrectly install or configure the TOE resulting in ineffective security mechanisms.	O.ADMIN_GUIDANCE The TOE will provide administrators with the necessary information for secure management.	O.ADMIN_GUIDANCE helps to mitigate this threat by ensuring the TOE administrators have guidance that instructs them how to administer the TOE in a secure manner. Having this guidance helps to reduce the mistakes that an administrator might make that could cause the TOE to be configured in a way that is insecure.

Threat/Policy	Objectives Addressing the Threat	Rationale
T.AUDIT_COMPROMISE A user or process may view audit records, cause audit records to be lost or modified, or prevent future audit records from being recorded, thus masking a user's action.	O.AUDIT_PROTECTION The TOE will provide the capability to protect audit information. O.RESIDUAL_INFORMATION The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated. O.SELF_PROTECTION The TSF will maintain a domain for its own execution that protects itself and its resources from external interference, tampering, or unauthorized disclosure through its own interfaces.	O.AUDIT_PROTECT contributes to mitigating this threat by controlling access to the audit trail. Only the System Administrator is allowed to read the audit trail, no one is allowed to modify audit records, the System Administrator is the only one allowed to delete the audit trail, and the TOE has the capability to prevent auditable actions from occurring if the audit trail is full. O.RESIDUAL_INFORMATION prevents a user not authorized to read the audit trail from access to audit information that might otherwise be persistent in a TOE resource (e.g., memory). By ensuring the TOE prevents residual information in a resource, audit information will not become available to any user or process except those explicitly authorized for that data. O. PARTIAL_SELF_PROTECTION contributes to countering this threat by ensuring that the TSF can protect itself from users. If the TSF could not maintain and control its domain of execution, it could not be trusted to control access to the resources under its control, which includes the audit trail which are always invoked is also critical to the mitigation of this threat.

The first two columns of this table are identical to the first two columns of Table 7. The rationale should address how each objective contributes to mitigating the threat or implementing the policy, and the applicable components from each objective should be identified. In Appendix A of this manual, we have supplied sample rational for several threats.

12. The PP authors should then write the objective/requirement component rationale. The format for this rationale should be as is shown in Table 6.

Table 6 Objective to Requirements Rationale

Objective	Requirements Addressing the Objective	Rationale

Objective	Requirements Addressing the Objective	Rationale
O.ADMIN_GUIDANCE The TOE will provide administrators with the necessary information for secure management.	ADO_DEL.1 ADO_IGS.1 AGD_ADM.1 AGD_USR.1 AVA_MSU.1	**ADO_DEL.1** ensures that the administrator is provided documentation that instructs them how to ensure the delivery of the TOE, in whole or in parts, has not been tampered with or corrupted during delivery. This requirement ensures the administrator has the ability to begin their TOE installation with a *clean* (e.g., malicious code has not been inserted once it has left the developer's control) version of the TOE, which is necessary for secure management of the TOE. **ADO_IGS.1** ensures the administrator has the information necessary to install the TOE in the evaluated configuration. Often times a vendor's product contains software that is not part of the TOE and has not been evaluated. The Installation, Generation and Startup (IGS) documentation ensures that once the administrator has followed the installation and configuration guidance the result is a TOE in a secure configuration. **AGD_ADM.1** mandates the developer provide the administrator with guidance on how to operate the TOE in a secure manner. This includes describing the interfaces the administrator uses in managing the TOE, security parameters that are configurable by the administrator, how to configure the TOE's rule set and the implications of any dependencies of individual rules. The documentation also provides a description of how to setup and review the auditing features of the TOE. **AGD_USR.1** is intended for non-administrative users, but could be used to provide guidance on security that is common to both administrators and non-administrators (e.g., password management guidelines). Since the non-administrative users of this TOE are limited to proxy users it is expected that the user guidance would discuss the secure use of proxies and how the single-use authentication mechanism is used. The use of the single-use authentication mechanism would not have to be repeated in the administrator's guide. AVA_MSU.1 ensures that the guidance documentation is complete and consistent, and notes all requirements for external security measures.

Objective	Requirements Addressing the Objective	Rationale
O.AUDIT_GENERATION The TOE will provide the capability to detect and create records of security-relevant events associated with users.	FAU_GEN.1-NIAP-0407 FAU_GEN.2-NIAP-0410 FIA_USB.1-NIAP-0415 FAU_SEL.1-NIAP-0407	**FAU_GEN.1-NIAP-0407** defines the set of events that the TOE must be capable of recording. This requirement ensures that the Administrator has the ability to audit any security relevant event that takes place in the TOE. This requirement also defines the information that must be contained in the audit record for each auditable event. This requirement also places a requirement on the level of detail that is recorded on any additional security functional requirements an ST author adds to this PP. **FAU_GEN.2-NIAP-0410** ensures that the audit records associate a user identity with the auditable event. In the case of authorized users, the association is accomplished with the userid. In all other cases, the association is based on the source network identifier, which is presumed to be the correct identity, but cannot be confirmed since these subjects are not authenticated. **FIA_USB.1-NIAP-0415** plays a role is satisfying this objective by requiring a binding of security attributes associated with users that are authenticated with the subjects that represent them in the TOE. This only applies to authorized users, since the identity of unauthenticated users cannot be confirmed. Therefore, the audit trail may not always have the proper identity of the subject that causes an audit record to be generated (e.g., presumed network address of an unauthenticated user may be a spoofed address). **FAU_SEL.1-NIAP-0407** allows the Security Administrator to configure which auditable events will be recorded in the audit trail. This provides the administrator with the flexibility in recording only those events that are deemed necessary by site policy, thus reducing the amount of resources consumed by the audit mechanism.

As with the previous rationale, the objective/component rationale should address how each component contributes to satisfying the objective. In Appendix B of this Manual, we have supplied sample rational for several objectives.

In writing the rationale sections the PP, authors may discover that a threat is not mitigated to the extent desired, or that an objective is not fully met. The PP authors will have to resolve these discrepancies by adjusting the threat/policy/objective statement or by adjusting component or element text, or by including a new component.

Table 7 Applicable Threats, Policies, Objectives, and Requirement Components for Basic Robustness PPs

Threat/Policy Basic	Objectives Addressing the Threat	Requirements associated with Objectives addressing the Threat	Notes
T.ACCIDENTAL_ADMIN_ERROR An administrator may incorrectly install or configure the TOE resulting in ineffective security mechanisms.	O.ADMIN_GUIDANCE The TOE will provide administrators with the necessary information for secure management.	ADO_DEL.1, ADO_IGS.1, AGD_ADM.1, AGD_USR.1, AVA_MSU.1	
T.ACCIDENTAL_AUDIT_COMPROMISE A user or process may view audit records, cause audit records to be lost or modified, or prevent future audit records from being recorded, thus masking a user's action.	O.AUDIT_PROTECTION The TOE will provide the capability to protect audit information.	FMT_MOF, FAU_SAR.2, FAU_STG.1-NIAP-0429, FAU_STG.3, FAU_STG.NIAP-0414-1	There should exist an iteration of FMT_MOF that applies to the audit functionality of the system; that iteration should be associated with this threat/objective combination. For FAU_STG.1-NIAP-0429 the PP authors should include the text in Instruction 17. For FAU_STG.3, the PP authors should include the text written in Instruction 18. FAU_STG.NIAP-0414-1 provides functionality similar to FAU_STG.4, the PP authors should include the text written in Instruction 19.
	O.RESIDUAL_INFORMATION The TOE will ensure that any information contained in a protected resource within its Scope of Control is not released when the resource is reallocated.	FDP_RIP.1	The PP authors should ensure the resources covered by FDP_RIP.1 include all of those in the TOE's scope of control.

Threat/Policy Basic	Objectives Addressing the Threat	Requirements associated with Objectives addressing the Threat	Notes
	O.SELF_PROTECTION The TSF will maintain a domain for its own execution that protects itself and its resources from external interference, tampering, or unauthorized disclosure through its own interfaces.	FPT_SEP.1, FPT_RVM.1	If the TOE is a software-only TOE, then the O.PARTIAL_SELF_PROTECTION objective (see Instruction 2) should be used. Also, FPT_SEP_EXP.1 should be used instead of FPT_SEP.1, and FPT_SEP_ENV_EXP.1 should be placed on the IT environment as outlined in Instruction 2. If the PP authors choose a higher level of FPT_SEP, then they should examine Instruction 2 for the FPT_SEP requirement to include.
T.ACCIDENTAL_ CRYPTO_ COMPROMISE A user or process may cause key, data or executable code associated with the cryptographic functionality to be inappropriately accessed (viewed, modified, or deleted), thus compromising the cryptographic mechanisms and the data protected by those mechanisms.	O.RESIDUAL_ INFORMATION The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated.	FCS_CKM	See Instruction 20 for cryptography in TOE in general.
T.MASQUERADE A user or process may masquerade as another entity in order to gain unauthorized access to data or TOE resources.	O.TOE_ACCESS The TOE will provide mechanisms that control a user's logical access to the TOE.	FIA_AFL.1-NIAP-0425, FIA_ATD.1, FIA_UID, FIA_UAU, AVA_SOF	This is an area that different technologies may address in different ways; some modification of the threat and objective may be necessary. The choice of the applicable FIA requirements will also depend on technology-specific concerns. For FIA_AFL.1-NIAP-0425 see Instruction 23 for suggested text.
T.POOR_DESIGN	O.CONFIGURATION_IDENTIFICATION	ACM_CAP.2, ALC_FLR.2	

Threat/Policy Basic	Objectives Addressing the Threat	Requirements associated with Objectives addressing the Threat	Notes
Unintentional errors in requirements specification or design of the TOE may occur, leading to flaws that may be exploited by a casually mischievous user or program.	The configuration of the TOE is fully identified in a manner that will allow implementation errors to be identified, corrected with the TOE being redistributed promptly.,		
	O.RATINGS_ MAINTENANCE Procedures to maintain the TOE's rating will be documented and followed.	AMA_AMP.1, AMA_CAT.1, AMA_EVD.1, AMA_SIA.1	
	O.DOCUMENTED_DESIGN The design of the TOE is adequately and accurately documented.	ADV_FSP.1, ADV_HLD.1, ADV_RCR.1	
	O.VULNERABILITY_ANALYSIS The TOE will undergo some vulnerability analysis demonstrate the design and implementation of the TOE does not contain any obvious flaws.	AVA_VLA.1	
T.POOR_IMPLEMENTATION Unintentional errors in implementation of the TOE design may occur, leading to flaws that may be exploited by a casually mischievous user or program.	O.CONFIGURATION_IDENTIFICATION The configuration of the TOE is fully identified in a manner that will allow implementation errors to be identified, corrected with the TOE being redistributed promptly.,	ACM_CAP.2, ALC_FLR.2	.
	O.RATINGS_MAINTENANCE Procedures to maintain the TOE's rating will be documented and followed.	AMA_AMP.1, AMA_CAT.1, AMA_EVD.1, AMA_SIA.1	
	O.PARTIAL_FUNCTIONAL_TESTING	ATE_COV.1, ATE_FUN.1, ATE_IND.2	

Threat/Policy Basic	Objectives Addressing the Threat	Requirements associated with Objectives addressing the Threat	Notes
	The TOE will undergo some security functional testing that demonstrates the TSF satisfies some of its security functional requirements.		
	O.VULNERABILITY_ANALYSIS The TOE will undergo some vulnerability analysis demonstrate the design and implementation of the TOE does not contain any obvious flaws.	AVA_VLA.1	
T.POOR_TEST Lack of or insufficient tests to demonstrate that all TOE security functions operate correctly (including in a fielded TOE) may result in incorrect TOE behavior being undiscovered thereby causing potential security vulnerabilities.	O.CORRECT_TSF_OPERATION The TOE will provide the capability to test the TSF to ensure the correct operation of the TSF at a customer's site.	FPT_TST.1	It appears that FPT_TST.1.1 refers to hardware functionality while FPT_TST.1.2 and FPT_TST.1.3 refer to the software. Additionally, certain types of TSF data (e.g., passwords, audit records) may prove troublesome with respect to FPT_TST.1.2 because they are dynamic. See Instruction 25 for further guidance.
	O.PARTIAL_FUNCTIONAL_TESTING The TOE will undergo some security functional testing that demonstrates the TSF satisfies the security functional requirements.	ATE_COV.1, ATE_FUN.1, ATE_IND.2	
	O.VULNERABILITY_ANALYSIS The TOE will undergo some vulnerability analysis to demonstrate the design and implementation of the TOE does not contain any obvious flaws.	AVA_VLA.1	
T.RESIDUAL_DATA	O.RESIDUAL_INFORMATION	FDP_RIP.1	

Threat/Policy Basic	Objectives Addressing the Threat	Requirements associated with Objectives addressing the Threat	Notes
A user or process may gain unauthorized access to data through reallocation of TOE resources from one user or process to another.	The TOE will ensure that any information contained in a protected resource within its Scope of Control is not released when the resource is reallocated.		
T.TSF_COMPROMISE A user or process may cause, through an unsophisticated attack,, TSF data, or executable code to be inappropriately accessed (viewed, modified, or deleted).	O.RESIDUAL_INFORMATION The TOE will ensure that any information contained in a protected resource within its Scope of Control is not released when the resource is reallocated.	FDP_RIP.1	
	O.SELF_PROTECTION The TSF will maintain a domain for its own execution that protects itself and its resources from external interference, tampering, or unauthorized disclosure through its own interfaces.	FPT_SEP.1, FPT_RVM.1	If the TOE is a software-only TOE, then the O.PARTIAL_SELF_PROTECTION objective (see Instruction 2) should be used. Also, FPT_SEP_EXP.1 should be used instead of FPT_SEP.1, and FPT_SEP_ENV_EXP.1 should be placed on the IT environment as outlined in Instruction 2. If the PP authors choose a higher level of FPT_SEP, then they should examine Instruction 2 for the FPT_SEP requirement to include.
	O.MANAGE	FMT_MTD.1, FMT_MSA.1, FMT_MOF.1	For MTD and MOF, the PP authors should group the data

Threat/Policy Basic	Objectives Addressing the Threat	Requirements associated with Objectives addressing the Threat	Notes
	The TOE will provide all the functions and facilities necessary to support the administrators in their management of the security of the TOE, and restrict these functions and facilities from unauthorized use.		and functions according to 1) who has access and 2) the actions that the users can perform. The requirements should be iterated for each unique set of actions that are specified. It should be noted that for FMT_MSA.1, the attributes are defined with respect to a user data access control policy (FDP_ACC, FDP_IFC) and should not be used for general "security attribute" restrictions.
T.UNATTENDED_ SESSION A user may gain unauthorized access to an unattended session.	O.TOE_ACCESS The TOE will provide mechanisms that control a user's logical access to the TOE.	FTA_SSL.1, FTA_SSL.2, FTA_SSL.3, AVA_SOF.1	FTA_SSL.3 is needed only if remote activity (e.g., remote administration) is included as required functionality for this technology.
T.UNAUTHORIZED_ ACCESS A user may gain access to user data for which they are not authorized according to the TOE security policy.	O.MEDIATE The TOE must protect user data in accordance with its security policy.	FDP_AC*, FDP_IF*	This threat is one of the most technology-specific, and will likely require substantial modification to focus on the access control policy implemented in the technology. This applies only to user data (TSF data are covered by other threats). Additional objectives may need to be created, and the wording for O.MEDIATE will likely need to be modified. It may not be necessary to include both the FDP_AC* and FDP_IF* families. Other components from FDP might also be included, again dependent on the technology. See Instruction 21 and 22 for usage of FDP_ACF and FDP_IFF, respectively, if chosen for the PP.
T.UNIDENTIFIED_ ACTIONS	O.AUDIT_REVIEW	FAU_SAR.1, FAU_SAR.3	For FAU_SAR.3, the first selection should be "searches

Threat/Policy Basic	Objectives Addressing the Threat	Requirements associated with Objectives addressing the Threat	Notes
The administrator may not have the ability to notice potential security violations, thus limiting the administrator's ability to identify and take action against a possible security breach.	The TOE will provide the capability to selectively view audit information,.		and sorting" to indicate that the capability to both search and to sort on the criteria is desired. The assignment in FAU_SAR.3 should include at least user identity, date, and time; technology-specific information should be included by the PP Authors in this list as well.
P.ACCESS_BANNER The TOE shall display an initial banner describing restrictions of use, legal agreements, or any other appropriate information to which users consent by accessing the system.	O.DISPLAY_BANNER The TOE will display an advisory warning regarding use of the TOE.	FTA_TAB.1	
P.ACCOUNTABILITY The authorized users of the TOE shall be held accountable for their actions within the TOE.	O.AUDIT_GENERATION The TOE will provide the capability to detect and create records of security-relevant events associated with users.	FAU_GEN.1-NIAP-0407, FAU_GEN.2-NIAP-0410, FIA_USB.1-NIAP-0415, FAU_SEL.1-NIAP-0407	FAU_GEN.1-NIAP-0407 and FAU_GEN.2-NIAP-410 should be included as indicated in Instruction 15; the audit event types and additional audit information should be included in a table and will specific to the requirements in the finalized PP. .See Instruction 24 for the suggested text for FIA_USB.1-NIAP-0415. For FAU_SEL.1, see Instruction 12.
	O.TIME_STAMPS The TOE shall provide reliable time stamps and the capability for the administrator to set the time used for these time stamps.	FPT_STM.1, FMT_MTD.1	There should be a FMT_MTD.1 iteration that covers setting the time that applies to this objective. FPT_STM.1 requires hardware to implement. If specifying a software-only TOE, FPT_STM.1 must be placed as a requirement on the IT environment. See Instruction 2 in this case.
	O.TOE_ACCESS	FIA_UID	

Threat/Policy Basic	Objectives Addressing the Threat	Requirements associated with Objectives addressing the Threat	Notes
	The TOE will provide mechanisms that control a user's logical access to the TOE.		
P.CRYPTOGRAPHY Only NIST FIPS validated cryptography (methods and implementations) are acceptable for key management (i.e.; generation, access, distribution, destruction, handling, and storage of keys) and cryptographic services (i.e.; encryption, decryption, signature, hashing, key exchange, and random number generation services).	O.CRYPTOGRAPHY The TOE shall use NIST FIPS 140-2 validated cryptographic services.		See Instruction 20 for a general discussion of cryptography
	O.RESIDUAL_INFORMATION The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated.		See Instruction 20 for a general discussion of cryptography

INSTRUCTION 10: SPECIFYING REQUIREMENTS ON THE IT ENVIRONMENT

(Back to TOC)

While especially relevant to software-only TOEs, PP authors for all TOEs should consider whether to include requirements on the IT Environment. Such requirements are often mis-specified or not included at all, even when it is appropriate. The PPRB recommends that such requirements be specified when appropriate, and offers the following guidance to PP authors in determining when they should specify requirements on the IT environment, and how they should specify those requirements.

In general, if a TOE depends upon another IT entity in order for the TOE to enforce its security policies, then IT environmental requirements are used to specify the behavior expected from that IT entity. Some PPs have attempted to use Assumptions (as in Instruction 7) to deal with the dependencies a TOE has on other IT products; this is an incorrect use of assumptions. Specifying IT environment requirements affords the PP author the opportunity to state what security functionality is required of other IT products using the same requirements language as that used to specify the TOE's security functionality. Using the same language is important because it allows the end user to more easily ascertain whether IT products can work together to enforce a security policy.

For example, if a database uses an operating system's files for storage and uses named pipes for inter-process communication then the database is relying on the operating system to protect those objects in order for the database to enforce its policies. The database PP author would then levy the FDP_ACC requirement on the IT environment (i.e., operating system) filling in the assignment for objects with files and named pipes. This would allow an end user who is attempting to compose a database with an operating system to determine if the operating system provides appropriate access control for the underlying database objects (e.g., files, named pipes).

When determining what requirements should be levied on the IT environment, the PP author considers what interaction the TOE will have with other IT entities and how that interaction may impact the TOE's ability to enforce its policies. If the TOE stores or obtains TSF data or security attributes from another IT entity, then the TOE has some security relevant dependency on that IT entity. If the TOE has a trust relationship with another IT entity, then the TOE probably has some dependency on that IT entity. The PP author considers the extent of the TOE's dependencies on that IT entity and determines what security functionality must be present in that IT entity to make it trustworthy from the perspective of the TOE.

One approach to determining the IT environment requirements would be to consider the IT entity as though it were part of the TOE. The PP author could then determine if the requirements levied on the TOE would apply to this "piece". The PP author then considers whether any additional requirements need to be specified on IT environment due to the nature of how the TOE depends on the trusted IT entity. Typically, if the TOE has the FPT_SEP requirement then the IT environment will have FPT_SEP levied against it as well (see Instruction 2 for more on FPT_SEP in the IT Environment). Another

requirement to consider is if the TOE requires communication channels (FTP_ITC) that are encrypted, then the IT environment requirements should levy the same requirements as are on the TOE, including the encryption that is required (i.e., the FCS family). Access control of objects that contain TSF data or security attributes should also be considered, as well as the accompanying FPT_RVM requirement.

With respect to presentation, when writing IT environment requirements the PP author should replace the text **TSF** with the text **IT environment**. This makes sense because the TSF is not ensuring the functionality; rather it is the IT environment that is expected to ensure the specified behavior. Other adjustments (e.g., replacing "TSC" with "IT environment's Scope of Control") may have to be made to the components as well.

INSTRUCTION 11: SCHEME INTERPRETATIONS

(Back to TOC)

This Consistency Instruction Manual requires that where applicable (e.g., for "new" PPs) NIAP Interpretations (NIs) and International Interpretations are used in developing PPs. Practical application of the CC and CEM against different types of security products and systems, as well as within different security environments, results in the need for interpretations of the CC and the CEM, in order to clarify their meaning.

As the Common Criteria is used by increasing numbers of people, inconsistencies or ambiguities are found in the wording. In order to address these concerns, the Common Criteria Interpretations Management Board (CCIMB) was formed. Regular meetings of the CCIMB, comprising representatives from the member nations, result in formal Interpretations, which specify textual updates to the CC and CEM.

National schemes likewise make pronouncements on any inconsistencies or ambiguities found in the CC, and may issue their own interpretations to be used within their own scheme; within CCEVS, the NIAP Interpretations Board (NIB) creates interpretations. NIAP, like all schemes, forwards its final interpretations to the CCIMB for international concurrence in order to minimize the divergence among the schemes. However, because the list of interpretations, both NIAP and international, is ever-increasing, it is impractical to attempt listing all final interpretations in this document; doing so would require constantly updating this document.

Within this document are some specific CC changes that the authors believe needed to be incorporated into PPs; these are presented as explicit requirements or refinements. Many of these suggested wording changes result from NIs, although many of these changes had not yet become international interpretations when this document was written. In such cases, within this document the PP author is reminded to check for an international interpretation that specifies the wording to be used, so that the new wording would not be considered an explicit requirement in need of justification.

If there is no international interpretation, then the PP author should check the NIs to see if there is specific wording supplied to be used within the PP; the rationale is simply that the new wording is the result of the NIAP interpretation.

Final International Interpretations can be found at:
http://www.commoncriteria.org/ccc/ri/finalIndex.jsp

Final NIAP Interpretations are available with other public NIB database entries at:
http://niap.nist.gov/cc-scheme/PUBLIC/index.html

INSTRUCTION 12: RATIONALE SECTION

(Back to TOC)

In this instruction the PPRB recommended that the PP authors spend a good deal of their effort in formulating detailed and comprehensive rationale. Writing rationale is sometimes difficult, but experience has shown that it is an important tool in producing high-quality PPs and offers the following points that PP authors should keep in mind while writing rationale.

The CC requires that a PP include rationale that demonstrates that the requirements satisfy the security objectives, and that those objectives counter the threats and implement the policies. The rationale serves two purposes. One purpose is to help the reader understand the intent of the requirements and objectives. The second purpose is that the process of writing a detailed rationale helps the PP author ensure that they have incorporated the appropriate requirements into the PP, and have made the proper selections and assignments within the requirements.

Since requirement language is written in English and typically consists of short concise statements, there is often room for interpretation. The PP author's intent may not be readily apparent in the requirements and they may be interpreted in a way that was not intended by the author. Having well-written rationale affords the PP author the opportunity to discuss what each requirement is attempting to achieve. The ultimate goal in writing a rationale is to communicate to the reader how the chosen requirements are intended to mitigate the associated threats, and implement the associated policies, and to what degree. Unfortunately, in an attempt to provide a different "view" of the system the CC includes the notion of security objectives, which provide a layer of indirection in achieving the ultimate goal of countering threats/implementing policies through requirements.

Requirements to Objective Rationale

One concern with the notion of security objectives is that currently a ST can claim conformance to a PP by demonstrating that the security objectives are satisfied. This means they do not necessarily have to include the same requirements. Since the objectives are also written in English and are usually written at a high general level, it leaves the security objectives open to interpretation and the result can be a PP conformant ST that does not meet the PP author's intent. By providing enough detail in the requirements to security objective rationale, the PP author can present the rationale in enough detail to ensure the intent of the objective is understood, making it more difficult for an ST author to claim conformance without satisfying the intent of the PP author. When writing the rationale that the requirements satisfy the objectives, the PP author should keep in mind the threats that are being addressed by the given objective and write the rationale for the requirements to security objectives so the reader can determine, in conjunction with the security objective to threat rationale to what degree the threats are being countered.

Objectives to Threat/Policy Rationale

When writing the security objectives to threat/policy rationale the PP author informs the reader to what extent a threat is being countered. The PP author should rely on the arguments made in the requirements to security objective rationale as the basis for making the argument that the threat is mitigated. It is acceptable, in fact desirable, to identify aspects of a threat that are not fully countered by the TOE. The threats provided in the PP guidance documents are somewhat generic and are written at a high level. The security objective to threat rationale should provide the details of what the TOE is protecting against. If there are technology specific aspects of the high level threats, then those specifics should be addressed in the rationale.

> For example, consider the T.MASQUERADE threat from Table 7: "A user or process may masquerade as another entity in order to gain unauthorized access to data or TOE resources." The authors of the Biometrics PP wanted to address several specific biometric-related threats in the PP, such as:

- an imposter may use an artificial hand/fingerprint or other synthetic means to gain unauthorized access;

- an imposter may know that their biometric characteristics are very similar to an enrollee and attempt to masquerade as that individual.

> Rather than creating several new threats, our recommended approach is to include T.MASQUERADE and O.TOE_ACCESS, and address these specific aspects of T.MASQUERADE in the rationale section for T.MASQUERADE to O.TOE_ACCESS.

Writing the security objective to threat rationale section is further complicated by the fact that typically more than one objective is used to mitigate a threat. In addition, different aspects of an objective may be used to mitigate different threats. This is because different requirements that are used to satisfy an objective are used to counter different threats. For **example**, the objective O.RESIDUAL_INFORMATION is satisfied by two requirements in the Medium Robustness Firewall PP: FDP.RIP.2 and FCS_CKM.4. The threat T.CRYPTO_COMPROMISE is partially mitigated by the objective O.RESIDUAL_INFORMATION, however, only the functionality provided by FCS_CKM.4 is discussed in the objective to threat rationale, since requirement ensures that cryptographic critical data will not be compromised by residing in resources that are not "cleaned" before being released to untrusted users. On the other hand, the threat T.AUDIT_COMPROMISE is partially mitigated by the objective O.RESIDUAL_INFORMATION, and only the functionality provided by FDP.RIP.2 is discussed in the rationale, FCS_CKM.4 does not contribute to satisfying the threat of a compromise of audit data occurring. To clarify exactly what is being addressed, the PPRB recommends that the requirement components applicable to a specific threat/policy be identified and associated with the objective; see the example of T.AUDIT_COMPROMISE in Table 5, Threats/Policy to Objective Rationale.

One of the reasons given above for writing good rationale is to help the PP author ensure they have included the appropriate CC components, and have made the appropriate assignments and selections within an element. When writing a PP, the author has a general idea of what family of requirements they want, but there may some indecision over the component that is chosen or what assignments and selections to make. Going through the exercise of making an argument of how and to what extent a threat is countered by a requirement or set of requirements forces the PP author to ensure they have the right requirements for what they are intending to protect against.

As an **example**, an early version of the firewall PP required functionality that locked a user's proxied session after a period of inactivity. The PP included FTA_SSL.1 and FTA_SSL.2 to mitigate the threat T.UNATTENDED_SESSION. These two components ensure that the user can initiate the locking of their session, and that after a time interval of inactivity the session is locked. After considering the threat and thinking how proxied sessions are used in a firewall, it was determined that these two components did not address remote sessions in a way that made sense. Therefore, FTA_SSL.3 was added, which requires that the remote session be terminated after a period of inactivity.

Assignments may not be filled in correctly, or there may be assignments that need to be made that aren't readily apparent. Writing good rationale can aid in identifying these areas as well. For example, the assignment of *time interval of inactivity* was modified in the FTA_SSL component. Originally this was left as an open assignment to be filled in by the ST author, which could have been any value the ST author deemed to be acceptable. After discussions about what was to be achieved with this requirement the assignment was changed to *administrator specified time period of inactivity*.

INSTRUCTION 13: CONVENTIONS

(Back to TOC)

Except for replacing United Kingdom spelling with American spelling, the notation, formatting, and conventions used in this PP are consistent with version 2.1 of the Common Criteria (CC). Selected presentation choices are discussed here to aid the PP reader.

The notation, formatting, and conventions used in this PP are largely consistent with those used in version 2.1 of the Common Criteria (CC). Selected presentation choices are discussed here to aid the PP user.
The CC allows several operations to be performed on functional requirements; *refinement*, *selection*, *assignment*, and *iteration* are defined in paragraph 2.1.4 of Part 2 of the CC. Each of these operations is used in this PP.

The **refinement** operation is used to add detail to a requirement, and thus further restricts a requirement. Refinement of security requirements is denoted by **bold text**.

The **selection** operation is used to select one or more options provided by the CC in stating a requirement. Selections that have been made by the PP authors are denoted by *italicized text*, selections to be filled in by the ST author appear in square brackets with an indication that a selection is to be made, [selection:], and are not italicized.

The **assignment** operation is used to assign a specific value to an unspecified parameter, such as the length of a password. Assignments that have been made by the PP authors are denoted by showing the value in square brackets, [Assignment_value], assignments to be filled in by the ST author appear in square brackets with an indication that an assignment is to be made [assignment:].

The **iteration** operation is used when a component is repeated with varying operations. Iteration is denoted by showing the iteration number in parenthesis following the component identifier, (iteration_number).

As this PP was sponsored, in part by NSA, National Information Assurance Partnership (NIAP) interpretations are used and are presented with the NIAP interpretation number as part of the requirement identifier (e.g., **FAU_GEN.1-NIAP-0407** for Audit data generation).

The CC paradigm also allows protection profile and security target authors to create their own requirements. Such requirements are termed ‘explicit requirements’ and are permitted if the CC does not offer suitable requirements to meet the authors’ needs. **Explicit requirements** must be identified and are required to use the CC class/family/component model in articulating the requirements. In this PP, explicit requirements will be indicated with the “(EXP)” following the component name. Application Notes are provided to help the developer, either to clarify the intent of a requirement, identify implementation choices, or to define “pass-fail” criteria for a

requirement. For those components where Application Notes are appropriate, the Application Notes will follow the requirement component.

NAMING CONVENTIONS

Assumptions: TOE security environment assumptions are given names beginning with "A." followed by a descriptive label all in caps -- e.g., A.ADMINISTRATION.

Threats: TOE security environment threats are given names beginning with "T." followed by a descriptive label all in caps-- e.g., T.SIGNAL_DETECT.

Policy Statements: Policy statements are given names beginning with "P." followed by a descriptive label all in caps-- e.g., P.PHYSICAL_ACCESS.

Security Objectives for the TOE: Security Objectives are given names beginning with "O." followed by a descriptive label all in caps-- e.g., O.ACCESS.

Security Objectives for both the IT Environment and Non-IT Environment: Security Objectives are given names beginning with "OE." followed by a descriptive label all in caps-- e.g., OE.ACCESS

INSTRUCTION 14: GLOSSARY

(Back to TOC)

The glossary is used to define very basic concepts such as roles and responsibilities that are specified in Protection Profiles (PPs) should be used consistently in all PPs. The independent definition and usage of redundant terms by multiple PP development teams leads to confusion amongst our target audiences of customers, vendors and evaluators.

The PPRB developed a set of term and definitions to be considered for inclusion in all PPs. The following list consists of terms that should be considered first by PP authors when trying to decide how best to describe their particular TOE and TOE environment. PP authors are dissuaded from developing new, redundant terminology and definitions when one of these terms may be adequate

Text

In the Common Criteria, many terms are defined in Section 2.3 of Part 1. The following are a subset of those definitions. They are listed here to aid the user of the PP being developed and should be included in the Glossary (Appendix B) of the Protection Profile.

Access -- Interaction between an entity and an object that results in the flow or modification of data.

Access Control -- Security service that controls the use of resources[4] and the disclosure and modification of data.[5]

Accountability -- Property that allows activities in an IT system to be traced to the entity responsible for the activity.

Administrator -- A user who has been specifically granted the authority to manage some portion or all of the TOE and whose actions may affect the TSP. Administrators may possess special privileges that provide capabilities to override portions of the TSP.

Assurance -- A measure of confidence that the security features of an IT system are sufficient to enforce its' security policy.

Asymmetric Cryptographic System -- A system involving two related transformations; one determined by a public key (the public transformation), and another determined by a private key (the private transformation) with the property that it is computationally infeasible to determine the private transformation (or the private key) from knowledge of the public transformation (and the public key).

---

[4] Hardware and software.

[5] Stored or communicated.

Asymmetric Key -- The corresponding public/private key pair needed to determine the behavior of the public/private transformations that comprise an asymmetric cryptographic system.

Attack -- An intentional act attempting to violate the security policy of an IT system.

Authentication -- Security measure that verifies a claimed identity.

Authentication data -- Information used to verify a claimed identity.

Authorization -- Permission, granted by an entity authorized to do so, to perform functions and access data.

Authorized user -- An authenticated user who may, in accordance with the TSP, perform an operation.

Availability -- Timely[6], reliable access to IT resources.

Compromise -- Violation of a security policy.

Confidentiality -- A security policy pertaining to disclosure of data.

Critical Security Parameters (CSP) -- Security-related information (e.g., cryptographic keys, authentication data such as passwords and pins, and cryptographic seeds) appearing in plaintext or otherwise unprotected form and whose disclosure or modification can compromise the security of a cryptographic module or the security of the information protected by the module.

Cryptographic Administrator -- An authorized user who has been granted the authority to perform cryptographic initialization and management functions. These users are expected to use this authority only in the manner prescribed by the guidance given to them.

Cryptographic boundary -- An explicitly defined contiguous perimeter that establishes the physical bounds (for hardware) or logical bounds (for software) of a cryptographic module.

Cryptographic key (key) -- A parameter used in conjunction with a cryptographic algorithm that determines [7]:

- the transformation of plaintext data into ciphertext data,
- the transformation of cipher text data into plaintext data,

[6] According to a defined metric.

- a digital signature computed from data,
- the verification of a digital signature computed from data, or
- a digital authentication code computed from data.

Cryptographic Module -- The set of hardware, software, firmware, or some combination thereof that implements cryptographic logic or processes, including cryptographic algorithms, and is contained within the cryptographic boundary of the module.

Cryptographic Module Security Policy -- A precise specification of the security rules under which a cryptographic module must operate, including the rules derived from the requirements of this PP and additional rules imposed by the vendor.

Defense-in-Depth (DID) -- A security design strategy whereby layers of protection are utilized to establish an adequate security posture for an IT system.

Discretionary Access Control (DAC) -- A means of restricting access to objects based on the identity of subjects and/or groups to which they belong. These controls are discretionary in the sense that a subject with a certain access permission is capable of passing that permission (perhaps indirectly) on to any other subject.

Embedded Cryptographic Module -- One that is built as an integral part of a larger and more general surrounding system (i.e., one that is not easily removable from the surrounding system).

Enclave -- A collection of entities under the control of a single authority and having a homogeneous security policy. They may be logical, or may be based on physical location and proximity.

Entity -- A subject, object, user or another IT device, which interacts with TOE objects, data, or resources.

External IT entity -- Any trusted Information Technology (IT) product or system, outside of the TOE, which may, in accordance with the TSP, perform an operation.

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

Integrity -- A security policy pertaining to the corruption of data and TSF mechanisms.

Integrity label -- A security attribute that represents the integrity level of a subject or an object. Integrity labels are used by the TOE as the basis for mandatory integrity control decisions.

Integrity level -- The combination of a hierarchical level and an optional set of non-hierarchical categories that represent the integrity of data.

Mandatory Access Control (MAC) -- A means of restricting access to objects based on subject and object sensitivity labels.[7]

Mandatory Integrity Control (MIC) -- A means of restricting access to objects based on subject and object integrity labels.

Multilevel -- The ability to simultaneously handle (e.g., share, process) multiple levels of data, while allowing users at different sensitivity levels to access the system concurrently. The system permits each user to access only the data to which they are authorized access.

Named Object -- An object that exhibits all of the following characteristics:

- The object may be used to transfer information between subjects of differing user identities within the TSF.
- Subjects in the TOE must be able to request a specific instance of the object.
- The name used to refer to a specific instance of the object must exist in a context that potentially allows subjects with different user identities to request the same instance of the object.

Non-Repudiation -- A security policy pertaining to providing one or more of the following:

- To the sender of data, proof of delivery to the intended recipient,
- To the recipient of data, proof of the identity of the user who sent the data.

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

Operating Environment -- The total environment in which a TOE operates. It includes the physical facility and any physical, procedural, administrative and personnel controls.

Operating System (OS) -- An entity within the TSC that causes operations to

[7] The Bell LaPadula model is an example of Mandatory Access Control

be performed. Subjects can come in two forms: trusted and untrusted. Trusted subjects are exempt from part or all of the TOE security policies. Untrusted subjects are bound by all TOE security policies.

Operational key -- Key intended for protection of operational information or for the production or secure electrical transmissions of key streams.

Peer TOEs -- Mutually authenticated TOEs that interact to enforce a common security policy.

Public Object -- An object for which the TSF unconditionally permits all entities "read" access. Only the TSF or authorized administrators may create, delete, or modify the public objects.

Robustness -- A characterization of the strength of a security function, mechanism, service or solution, and the assurance (or confidence) that it is implemented and functioning correctly. DoD has three levels of robustness:

> **Basic:** Security services and mechanisms that equate to good commercial practices.
>
> **Medium:** Security services and mechanisms that provide for layering of additional safeguards above good commercial practices.
>
> **High:** Security services and mechanisms that provide the most stringent protection and rigorous security countermeasures.

Secure State -- Condition in which all TOE security policies are enforced.

Security attributes -- TSF data associated with subjects, objects, and users that are used for the enforcement of the TSP.

Security level -- The combination of a hierarchical classification and a set of non-hierarchical categories that represent the sensitivity of the information [10].

Sensitivity label -- A security attribute that represents the security level of an object and that describes the sensitivity (e.g. Classification) of the data in the object. Sensitivity labels are used by the TOE as the basis for mandatory access control decision.

Split key -- A variable that consists of two or more components that must be combined to form the operational key variable. The combining process excludes concatenation or interleaving of component variables.

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

Symmetric key -- A single, secret key used for both encryption and decryption

in symmetric cryptographic algorithms.

Threat -- Capabilities, intentions and attack methods of adversaries, or any circumstance or event, with the potential to violate the TOE security policy.

Threat Agent - Any human user or Information Technology (IT) product or system, which may attempt to violate the TSP and perform an unauthorized operation with the TOE.

User -- Any entity (human user or external IT entity) outside the TOE that interacts with the TOE.

Vulnerability -- A weakness that can be exploited to violate the TOE security policy.

IV. Minimum Common Criteria Security Functional Requirement

(Back to TOC)

A. Security Audit

(Back to TOC)

Instruction 15: FAU_GEN.1-NIAP-0407 Audit data generation

FAU_GEN.2 USER IDENTITY ASSOCIATION

(Back to TOC)

The FAU_GEN.1-NIAP-0407 component should be structured in a consistent way. The events to be audited, as well as the information to be contained in the events, are currently presented in a variety of different ways. Further, the requirements as written may allow an ST writer to add components and not require auditing on the functionality provided by these components if the FAU_GEN.1-NIAP-0407 elements are used directly as indicated in the CC. Also, the FAU_GEN.2-NIAP-0410 component should be included as stated in the interpretation.

Therefore, the PPRB recommends the following standard wording and format (including the table) be used when FAU_GEN.1-NIAP-0407 and FAU_GEN.2-NIAP-0410 are included in the PP. The table in FAU_GEN.1-NIAP-0407 is for illustrative purposes only; the PP writing team should detail audit information is required for their PP.

When constructing the table, the PP authors should consider the "Basic" level of audit the starting point for selecting the events to be audited. However, when examining the Basic level of audit for each component included in the PP, the PP authors may choose to either omit or add events. The PP authors should examine other Basic Robustness PPs to determine in what instances strict adherence to the CC Basic level of audit may not be appropriate.

Suggested Text

FAU_GEN.1-NIAP-0407 Audit data generation

FAU_GEN.1.1-NIAP-0407 – 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 listed in Table 1;

c) [selection: [assignment: *events at a basic level of audit introduced by the inclusion of additional SFRs determined by the ST author*], [assignment: *events commensurate with a basic level of audit introduced by the inclusion of explicit requirements determined by the ST author*], "*no additional events*"].

Application Note: For the selection, the ST author should choose one or both of the assignments (as detailed in the following paragraphs), or select "no additional events".

For the first assignment, the ST author augments the table (or lists explicitly) the audit events associated with the basic level of audit for any SFRs that the ST author includes that are not included in this PP.

Likewise, for the second assignment the ST author includes audit events that may arise due to the inclusion of any explicit requirements not already in the PP. Because "basic" audit is not defined for such requirements, the ST author will need to determine a set of events that are commensurate with the type of information that is captured at the basic level for similar requirements.

If no additional (CC or explicit) SFRs are included, or if additional SFRs are included that do not have "basic" audit associated with them, then it is acceptable to assign "no additional events" in this item..

FAU_GEN.1.2-NIAP-0410 - 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 (if applicable), 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, [information specified in column three of Table 1 below].

Appl*ication Note: In column 3 of the table below, "if applicable" is used to designate data that should be included in the audit record if it "makes sense" in the context of the event that generates the record.. If no other information is required (other than that listed in "a") for a particular audit event type, then an assignment of "none" is acceptable.*

Table 8 – Auditable Events

Requirement	Auditable Events	Additional Audit Record Contents
FAU_GEN.1-NIAP-0407	None	
FAU_SAR.1	Opening the audit trail	The identity of the Audit Administrator performing the function
FAU_SAR.2	Unsuccessful attempts to read information from the audit records	The identity of the administrator performing the function

Requirement	Auditable Events	Additional Audit Record Contents
FAU_SAR.3	None	
FAU_SEL.1-NIAP-0407	All modifications to the audit configuration that occur while the audit collection functions are operating	The identity of the Security Administrator performing the function
(…all components in the PP should be included in this table…)		

FAU_GEN.2-NIAP-0410 User identity association

FAU_GEN.2.1-NIAP-0410 For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.

INSTRUCTION 16: FAU_SEL.1-NIAP-0407 AUDIT EVENT SELECTION

(Back to TOC)

The following text reflects the consistent selections and assignments that the PPRB recommends for all Basic Robustness PPs. PP authors should also include other technology-specific attributes on which to base the selectivity of audit.

Suggested Text

FAU_SEL.1-NIAP-0407 Selective Audit

FAU_SEL.1.1-NIAP-0407 - **Refinement:** The TSF shall **allow only the administrator** to include or exclude auditable events from the set of audited events based on the following attributes:

a) user identity;

b) event type;

c) [*selection: object identity, subject identity, host identity, "none"]*;

d) success of auditable security events;

e) failure of auditable security events; and

f) [*selection: [assignment: list of additional criteria that audit selectivity is based upon*], *no additional criteria*]].

Application Note: "event type" is to be defined by the ST author; the intent is to be able to include or exclude classes of audit events.

INSTRUCTION 17: FAU_STG.1-NIAP-0429 AUDIT EVENT STORAGE

(Back to TOC)

The PPRB recommends that the administrative role allowed to delete audit records be specifically specified in the requirement, and that modifications to the audit records in the audit trail be prevented. In order to implement these changes, as well as the interpretations to the FAU_STG.1 requirement, the following text and format should be used for Basic Robustness PPs.

Note that I-0423 changes FAU_STG.1.2 from "modifications" to "unauthorized modifications"; the PPRB recommends that *all* modifications (whether authorized or not) be prevented, thus the refinement for FAU_STG.1.2-NIAP-0429 below is suggested.

Suggested Text:

FAU_STG.1-NIAP-0429 Protected audit trail storage

FAU_STG.1.1-NIAP-0429 – **Refinement:** The TSF shall **restrict the deletion of stored** audit records in the audit trail **to the administrator**.

FAU_STG.1.2-NIAP-0429 **Refinement:** The TSF shall be able to *prevent* modifications to the audit records in the audit trail**.**

INSTRUCTION 18: FAU_STG.3 AUDIT EVENT STORAGE

(Back to TOC)

Should the PP author invoke FAU_STG.3, it should be structured in a common manner to reflect the same assignments across all Basic Robustness PPs.

This requirement calls for the percentage of the storage capacity to be administrator settable; this implies that an FMT_MOF or FMT_MTD requirement is needed as well. PP Authors should ensure that it is included when this component is included.

Suggested Text

FAU_STG.3 Action in case of possible audit data loss

FAU_STG.3.1 - **Refinement**: The TSF shall [immediately alert the administrators by displaying a message at the local console, [selection: [assignment: other actions determined by the ST author], "none"]] if the audit trail exceeds [an Administrator-settable percentage of storage capacity].

Application Note: The ST Author should determine if there are other actions that should be taken when the audit trial setting is exceeded, and put these in the assignment. If there are no other actions, then the ST Author should select "none".

INSTRUCTION 19: FAU_STG.NIAP-0414 AUDIT EVENT STORAGE[8]

(Back to TOC)

The PPRB recommends that the PP author specify functionality for audit trail loss for Basic Robustness PPs. Since it is desirable that this capability be administrator-settable, FAU_STG.NIAP-0429-1 should be used as follows.

FAU_STG.NIAP-0429-1 calls for the selection of the option taken by the administrator when there's an audit storage failure. The inclusion of requirement in the PP implies that an FMT_MOF or FMT_MTD requirement is needed as well. PP Authors should ensure that it is included when this component is included. If there are "special" administrators that are able to perform this function, then the application note and the text of the requirement should be changed as well.

Suggested Text

FAU_STG.NIAP-0414-1 Site-configurable Prevention of audit data loss

FAU_STG.NIAP-0414-1.1 The TSF shall provide an authorized administrator with the capability to select one or more of the following actions [*prevent auditable events, except those taken by the authorised user with special rights, overwrite the oldest stored audit records*] and [selection: [assignment: other actions to be taken in case of audit storage failure], "no additional options"] to be taken if the audit trail is full.

FAU_STG.NIAP-0414-2-NIAP-0429 The TSF shall [*selection: choose one of: "ignore auditable events", "prevent auditable events, except those taken by the authorized user with special rights", "overwrite the oldest stored audit records"*] and [*assignment: other actions to be taken in case of audit storage failure*] if the audit trail is full.

Application Note: The TOE provides the administrator the option of preventing audit data loss by preventing auditable events from occurring. The administrator's actions under these circumstances are not required to be audited. The TOE also provides the administrator the option of overwriting "old" audit records rather than preventing auditable events, which may protect against a denial-of-service attack.

The ST writer should fill in other technology-specific actions that can be taken for audit storage failure (in addition to the two already specified), or select "no additional options" if there are no such technology-specific actions.

---

[8] Interpretations I-0407 and I-0429 conflict in labeling this requirement because they each add their specific modification without regard for the other. The text in this document has been modified to take into account the changes to both I-0407 and I-0429, and the label has been chosen as "NIAP-0429".

B. CRYPTOGRAPHIC SUPPORT

(Back to TOC)

INSTRUCTION 20: FIPS140-2 (SECURITY REQUIREMENTS FOR CRYPTOGRAPHIC MODULES)

FCS_CKM Cryptographic Key Management
FCS_COP Cryptographic operation

(Back to TOC)

The TSF may employ cryptographic functionality to help satisfy several high-level security objectives. These include (but are not limited to): identification and authentication, non-repudiation, trusted path, trusted channel and data separation.

Cryptographic services might be provided in hardware or software, and might be provided at any level from link up through application. Cryptography might be based upon public-keys or on private key exchanges, and is implemented using any of a variety of algorithms, some of which can be certified under validation programs such as Federal Information Processing Standard (FIPS). Basic Robustness Protection Profile mandates the use of good commercial practices[9] and the use of FIPS140-2 (*Security Requirements for Cryptographic Modules) (*http://csrc.nist.gov/cryptval/140-2.htm*)* validated cryptography. Detailed cryptographic support for the common criteria cryptographic support elements (i.e., FCS_CKM, FCS_COP) should be provided by the Protection Profile development team's cryptographic support member/s. The cryptographic support organization may review the cryptographic support section of the protection profile for approval (this review and/or approval will be at the discretion of the cryptographic support organizations).

[9] Best commercial practices and processes are those practices and processes used by commercial industry that, over time, have proven cost effective, efficient and successful in bringing quality products to the marketplace.

C. User Data Protection

(Back to TOC)

Instruction 21: FDP_ACF Access control functions

(BACK TO TOC)

If the PP authors choose to use the FDP_ACF family requirements, they should use the following interpreted requirement text as a basis.

Interpreted Text:

FDP_ACF.1-NIAP-0407 Security attribute based access control

FDP_ACF.1.1-NIAP-0407: The TSF shall enforce the [assignment: access control SFP] to objects based on the following: [assignment: list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes]

FDP_ACF.1.2-NIAP-0407 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*].

FDP_ACF.1.3-NIAP-0407 The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [selection: [assignment: *rules, based on security attributes, that explicitly authorise access of subjects to objects*], "*no additional rules*"].

FDP_ACF.1.4-NIAP-0407 The TSF shall explicitly deny access of subjects to objects based on the [selection: [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*], "*no additional rules*"].

INSTRUCTION 22: FDP_IFF.1 AND .2 INFORMATION FLOW CONTROL FUNCTIONS

(Back to TOC)

If the PP authors choose to use the FDP_IFF.1 or .2 components, they should use the following interpreted requirement text as a basis.

Suggested Text:

FDP_IFF.1-NIAP-0407 Simple security attributes

FDP_IFF.1.1-NIAP-0407: The TSF shall enforce the [assignment: information flow control SFP] based on the following types of subject and information security attributes: [assignment: the minimum number and type of security attributes list of subjects and information controlled under the indicated SFP, and for each, the SFP-relevant security attributes]

FDP_IFF.1.2-NIAP-0407 The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: [assignment: *for each operation, the security attribute-based relationship that must hold between subject and information security attributes*].

FDP_IFF.1.3-NIAP-0407 The TSF shall enforce the following information flow control rules: [selection: [assignment: additional information flow control SFP rules], "no additional information flow control SFP rules"]

FDP_IFF.1.4-NIAP-0407 The TSF shall provide the following [selection: [assignment: list of additional SFP capabilities], "no additional SFP capabilities"]

FDP_IFF.1.5-NIAP-0407 The TSF shall explicitly authorise an information flow based on the following rules: [selection: [assignment: rules, based on security attributes, that explicitly authorise information flows], "no explicit authorisation rules"]

FDP_IFF.1.6-NIAP-0407 The TSF shall explicitly deny an information flow based on the following rules: [selection: [assignment: rules, based on security attributes, that explicitly deny information flows], "no explicit denial rules"]

FDP_IFF.2-NIAP-0407 Hierarchical security attributes

FDP_IFF.2.1-NIAP-0407: The TSF shall enforce the [assignment: information flow control SFP] based on the following types of subject and information security attributes: [assignment: the minimum number and type of security attributes list of subjects and information controlled under the indicated SFP, and for each, the SFP-relevant security attributes]

FDP_IFF.2.2-NIAP-0407 The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules, based on the ordering relationships of security attributes, hold: [assignment: *for each operation, the security attribute-based relationship that must hold between subject and information security attributes*].

FDP_IFF.2.3-NIAP-0407 The TSF shall enforce the following information flow control rules: [selection: [assignment: additional information flow control SFP rules], "no additional information flow control SFP rules"]

FDP_IFF.2.4-NIAP-0407 The TSF shall provide the following [selection: [assignment: list of additional SFP capabilities], "no additional SFP capabilities"]

FDP_IFF.2.5-NIAP-0407 The TSF shall explicitly authorise an information flow based on the following rules: [selection: [assignment: rules, based on security attributes, that explicitly authorise information flows], "no explicit authorisation rules"]

FDP_IFF.2.6-NIAP-0407 The TSF shall explicitly deny an information flow based on the following rules: [selection: [assignment: rules, based on security attributes, that explicitly deny information flows], "no explicit denial rules"]

FDP_IFF.2.7-NIAP-0407 The TSF shall enforce the following relationships for any two valid information flow control security attributes:

a) There exists an ordering function that, given two valid security attributes, determines if the security attributes are equal, if one security attribute is greater than the other, or if the security attributes are incomparable; and

b) There exists a “least upper bound” in the set of security attributes, such that, given any two valid security attributes, there is a valid security attribute that is greater than or equal to the two valid security attributes; and

c) There exists a “greatest lower bound” in the set of security attributes, such that, given any two valid security attributes, there is a valid security attribute that is not greater than the two valid security attributes.

D. IDENTIFICATION AND AUTHENTICATION

(Back to TOC)

INSTRUCTION 23: FIA_AFL.1-NIAP-0425 AUTHENTICATION FAILURES

(Back to TOC)

The PPRB recommends that authentication failure controls be present on all Basic Robustness PPs, and further that these controls be administrator-settable. The PPRB recommends the following text be included to capture this functionality for all Basic Robustness PPs.

Suggested Text:

FIA_AFL.1-NIAP-0425 Authentication failure handling

FIA_AFL.1.1-NIAP-0425: **Refinement:** The TSF shall detect when [an administrator configurable integer] **of** unsuccessful authentication attempts occur related to [assignment: list of authentication events].

FIA_AFL.1.2-NIAP-0425 When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF shall [prevent the [assignment: *entities requesting authentication*] from performing activities that require authentication until an action is taken by the administrator].

The PP authors should ensure that when the **entities requesting authentication** *is specified in the PP, at least one account should be exempted from the requirement so as to avoid an administrative denial of service.*

INSTRUCTION 24: FIA_USB.1 USER-SUBJECT BINDING

(Back to TOC)

In Table 7 the PPRB suggests including FIA_USB.1-NIAP-0415. This text is included below to capture the notion that all of the user attributes specified in FIA_ATD should be associated with subjects.

If the PP authors wish to specify rules governing the binding of users to subjects (which is not able to be specified using FIA_USB.1-NIAP-0415), the Interpreted Text below should be used as the template.

Suggested Text:

FIA_USB.1-NIAP-0415 User-Subject Binding

FIA_USB.1.1-NIAP-0415 - Refinement: The TSF shall associate all user security attributes with subjects acting on behalf of that user.

Interpreted Text:

FIA_USB.NIAP-0352-1: Expanded user-subject binding

FIA_USB.NIAP-0352-1.1: **Refinement:** The TSF shall associate **all** user security attributes with subjects acting on the behalf of that user.

FIA_USB.NIAP-0352-1.2: The TSF shall enforce the following rules on the initial association of user security attributes with subjects acting on the behalf of users: [assignment: initial association rules].

FIA_USB.NIAP-0352-1.3: The TSF shall enforce the following rules governing changes to the user security attributes associated with subjects acting on the behalf of users: [assignment: changing of attributes rules].

E. PROTECTION OF THE TSF

(Back to TOC)

INSTRUCTION 25: FPT_TST TSF SELF TEST

(BACK TO TOC)

The PPRB recommends that TSF testing be specified in all Basic Robustness PPs in order to validate aspects of the TSF prior to or while it is operating. However, there are two issues with FPT_TST.1 as it appears in the Common Criteria. First, the wording of FPT_TST.1.1 appears to make sense only if the TOE includes hardware; it is difficult to imagine what software TSF "self-tests" would be run. Secondly, some TOE data are dynamic (e.g., data in the audit trail, passwords) and so interpretation of "integrity" for FPT_TST.1.2 is required, leading to potential inconsistencies amongst Basic Robustness TOEs. The PPRB therefore makes the following two suggestions; the first for software-only TOEs, and the second for TOEs that include the hardware.

Suggested Text for Software-Only TOEs:

FPT_TST_EXP1.1 TSF testing

FPT_TST_EXP1.1.1 - The TSF shall provide administrator with the capability to verify the integrity of the following TSF data: [assignment: TSF data for which integrity validation is required].

FPT_TST_EXP1.1.2 - The TSF shall provide administrator with the capability to verify the integrity of stored TSF executable code.

Suggested Text for TOEs That Include Hardware:

FPT_TST_EXP2.1 TSF testing

FPT_TST_EXP2.1.1 – The TSF shall run a suite of self-tests *during initial start-up, periodically during normal operation as specified by the administrator*, and at the [request of an administrator] to demonstrate the correct operation of the TSF.

FPT_TST_EXP2.1.2 - The TSF shall provide administrator with the capability to verify the integrity of the following TSF data: [assignment: TSF data for which integrity validation is required].

FPT_TST_EXP2.1.3 - The TSF shall provide administrator with the capability to verify the integrity of stored TSF executable code.

V. Appendices

(Back to TOC)

APPENDIX A: MAPPING OF BASIC ROBUSTNESS THREATS/POLICIES TO OBJECTIVES

(Back to TOC)

Sample rationale is provided below. The PP authors should examine various NIAP evaluated PPs for examples of rationale.

Threat/Policy	Objectives Addressing the Threat and Policies	Rationale
T.ACCIDENTAL_ADMIN_ERROR: An administrator may incorrectly install or configure the TOE resulting in ineffective security mechanisms.	**O.ADMIN_GUIDANCE**: The TOE will provide administrators with the necessary information for secure management.	**O.ADMIN_GUIDANCE** helps to mitigate this threat by ensuring the TOE administrators have guidance that instructs them how to administer the TOE in a secure manner. Having this guidance helps to reduce the mistakes that an administrator might make that could cause the TOE to be configured in a way that is insecure.
T.ACCIDENTAL_AUDIT_COMPROMISE: A user or process may view audit records, cause audit records to be lost or modified, or prevent future audit records from being recorded, thus masking a user's action.	**O.AUDIT_PROTECTION**: The TOE will provide the capability to protect audit information. **O.RESIDUAL_ INFORMATION**: The TOE will ensure that any information contained in a protected resource within its Scope of Control is not released when the resource is reallocated. **O.SELF_PROTECTION**: The TSF will maintain a domain for its own execution that protects itself and its resources from external interference, tampering, or unauthorized disclosure through its own interfaces.	**O.AUDIT_PROTECT** contributes to mitigating this threat by controlling access to the audit trail. Only the System Administrator is allowed to read the audit trail, no one is allowed to modify audit records, the System Administrator is the only one allowed to delete the audit trail, and the TOE has the capability to prevent auditable actions from occurring if the audit trail is full. **O.RESIDUAL_INFORMATION** prevents a user not authorized to read the audit trail from access to audit information that might otherwise be persistent in a TOE resource (e.g., memory). By ensuring the TOE prevents residual information in a resource, audit information will not become available to any user or process except those explicitly authorized for that data. **O.PARTIAL_SELF_PROTECTION** contributes to countering this threat by ensuring that the TSF can protect itself from users. If the TSF could not maintain and control its domain of execution, it could not be trusted to control access to the resources under its control, which includes the audit trail which are always invoked is also critical to the migration of this threat.

Threat/Policy	Objectives Addressing the Threat and Policies	Rationale
T.ACCIDENTAL_ CRYPTO_ COMPROMISE: A user or process may cause key, data or executable code associated with the cryptographic functionality to be inappropriately accessed (viewed, modified, or deleted), thus compromising the cryptographic mechanisms and the data protected by those mechanisms.	**O.RESIDUAL_ INFORMATION**: The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated.	**O.RESIDUAL_INFORMATION** counters this threat by ensuring that TSF data and user data is not persistent when resources are released by one user/process and allocated to another user/process
T.MASQUERADE: A user or process may masquerade as another entity in order to gain unauthorized access to data or TOE resources.	**O.TOE_ACCESS**: The TOE will provide mechanisms that control a user's logical access to the TOE.	**O.TOE_ACCESS** mitigates this threat by controlling the logical access to the TOE and its resources. By constraining how and when authorized users can access the TOE, and by mandating the type and strength of the authentication mechanism this objective helps mitigate the possibility of a user attempting to login and masquerade as an authorized user. In addition, this objective provides the administrator the means to control the number of failed login attempts a user can generate before an account is locked out, further reducing the possibility of a user gaining unauthorized access to the TOE.
T.POOR_DESIGN: Unintentional errors in requirements specification or design of the TOE may occur, leading to flaws that may be exploited by a casually mischievous user or program.	**O.CONFIGURATION_IDENTIFICATION**: The configuration of the TOE is fully identified in a manner that will allow implementation errors to be identified, corrected with the TOE being redistributed promptly, **O.DOCUMENTED_DESIGN**: The design of the TOE is adequately and accurately documented. **O.VULNERABILITY_ANALYSIS**: The TOE will undergo some vulnerability analysis to demonstrate the design and implementation of the TOE does not contain any obvious flaws.	**O.CONFIGURATION_IDENTIFI-CATION** plays a role in countering this threat by requiring the developer to provide control of the changes made to the TOE's design. **O.DOCUMENTED_DESIGN** ensures that the design of the TOE is documented, permitting detailed review by evaluators and validators. **O.VULNERABILITY_ANALYSIS_-TEST** ensures that the design of the TOE is analyzed for design flaws.

Threat/Policy	Objectives Addressing the Threat and Policies	Rationale
T.POOR_IMPLEMENTATION: Unintentional errors in implementation of the TOE design may occur, leading to flaws that may be exploited by a casually mischievous user or program.	**O.CONFIGURATION_IDENTIFICATION**: The configuration of the TOE is fully identified in a manner that will allow implementation errors to be identified, corrected with the TOE being redistributed promptly., **O.PARTIAL_FUNCTIONAL_TESTING**: The TOE will undergo some security functional testing that demonstrates the TSF satisfies some of its security functional requirements. **O.VULNERABILITY_ANALYSIS**: The TOE will undergo some vulnerability analysis demonstrate the design and implementation of the TOE does not contain any obvious flaws.	**O. CONFIGURATION_IDENTIFICATION** plays a role in countering this threat by requiring the developer to provide control of the changes made to the TOE's design. Although the previous three objectives help minimize the introduction of errors into the implementation, **O.PARTIAL_FUNCTIONAL_TEST-ING** increases the likelihood that any errors that do exist in the implementation (with respect to the functional specification, high level, and low-level design) will be discovered through testing. **O.VULNERABILITY_ANALYSIS_-TEST** helps reduce errors in the implementation that may not be discovered during functional testing. Ambiguous design documentation, and the fact that exhaustive testing of the external interfaces is not required may leave bugs in the implementation undiscovered in functional testing

Threat/Policy	Objectives Addressing the Threat and Policies	Rationale
T.POOR_TEST: Lack of or insufficient tests to demonstrate that all TOE security functions operate correctly (including in a fielded TOE) may result in incorrect TOE behavior being undiscovered thereby causing potential security vulnerabilities.	**O.DOCUMENTED_DESIGN** The design of the TOE will be adequately and accurately documented. **O.CORRECT_ TSF_OPERATION**: The TOE will provide the capability to test the TSF to ensure the correct operation of the TSF at a customer's site. **O.PARTIAL_FUNCTIONAL_TESTING**: The TOE will undergo some security functional testing that demonstrates the TSF satisfies the security functional requirements. **O.VULNERABILITY_ANALYSIS**: The TOE will undergo some vulnerability analysis demonstrate the design and implementation of the TOE does not contain any obvious flaws.	**O.DOCUMENTED_DESIGN** helps to ensure that the TOE's documented design satisfies the security functional requirements. In order to ensure the TOE's design is correctly realized in its implementation, the appropriate level of functional testing of the TOE's security mechanisms must be performed during the evaluation of the TOE. **O.CORRECT_ TSF_OPERATION** ensures that once the TOE is installed at a customer's location, the capability exists that the integrity of the TSF (hardware and software) can be demonstrated, and thus providing end users the confidence that the TOE's security policies continue to be enforced. **O.PARTIAL_FUNCTIONAL_TEST-ING** increases the likelihood that any errors that do exist in the implementation (with respect to the functional specification, high level, and low-level design) will be discovered through testing. **O.VULNERABILITY_ANALYSIS_-TEST** addresses this concern by requiring a vulnerability analysis be performed in conjunction with testing that goes beyond functional testing. This objective provides a measure of confidence that the TOE does not contain security flaws that may not be identified through functional testing. While these testing activities are a necessary activity for successful completion of an evaluation, this testing activity does not address the concern that the TOE continues to operate correctly and enforce its security policies once it has been fielded. Some level of testing must be available to end users to ensure the TOE's security mechanisms continue to operate correctly once the TOE is fielded

Threat/Policy	Objectives Addressing the Threat and Policies	Rationale
T.RESIDUAL_DATA: A user or process may gain unauthorized access to data through reallocation of TOE resources from one user or process to another.	**O.RESIDUAL_INFORMATION**: The TOE will ensure that any information contained in a protected resource within its Scope of Control is not released when the resource is reallocated.	**O.RESIDUAL_INFORMATION** counters this threat by ensuring that TSF data and user data is not persistent when resources are released by one user/process and allocated to another user/process.
T.TSF_COMPROMISE: A user or process may cause, through an unsophisticated attack,, TSF data, or executable code to be inappropriately accessed (viewed, modified, or deleted).	**O.RESIDUAL_INFORMATION**: The TOE will ensure that any information contained in a protected resource within its Scope of Control is not released when the resource is reallocated. **O.PARTIAL_SELF_PROTECTION**: The TSF will maintain a domain for its own execution that protects itself and its resources from external interference, tampering, or unauthorized disclosure through its own interfaces. **O.MANAGE**: The TOE will provide all the functions and facilities necessary to support the administrators in their management of the security of the TOE, and restrict these functions and facilities from unauthorized use.	**O.RESIDUAL_INFORMATION** is necessary to mitigate this threat, because even if the security mechanisms do not allow a user to explicitly view TSF data, if TSF data were to inappropriately reside in a resource that was made available to a user, that user would be able to inappropriately view the TSF data. **O.PARTIAL_SELF_PROTECTION**: The TSF will maintain a domain for its own execution that protects itself and its resources from external interference, tampering, or unauthorized disclosure through its own interfaces. **O.MANAGE** is necessary because an access control policy is not specified to control access to TSF data. This objective is used to dictate who is able to view and modify TSF data, as well as the behavior of TSF functions.
T.UNATTENDED_ SESSION: A user may gain unauthorized access to an unattended session.	**O.TOE_ACCESS**: The TOE will provide mechanisms that control a user's logical access to the TOE.	**O.TOE_ACCESS** helps to mitigate this threat by including mechanisms that place controls on user's sessions. Local administrator's sessions are locked and remote sessions are dropped after a Security Administrator defined time period of inactivity. Locking the local administrator's session reduces the opportunity of someone gaining unauthorized access the session when the console is unattended. Dropping the connection of a remote session (after the specified time period) reduces the risk of someone accessing the remote machine where the session was established, thus gaining unauthorized access to the session

Threat/Policy	Objectives Addressing the Threat and Policies	Rationale
T.UNAUTHORIZED_ ACCESS: A user may gain access to user data for which they are not authorized according to the TOE security policy.	**O.MEDIATE**: The TOE must protect user data in accordance with its security policy.	**O.MEDIATE** ensures that all accesses to user data are subject to mediation, unless said data has been specifically identified as public data**.** The TOE requires successful authentication to the TOE prior to gaining access to any controlled-access content. By implementing strong authentication to gain access to these services, an attacker's opportunity to successfully conduct a man-in-the-middle and/or password guessing attack is greatly reduced. Lastly, the TSF will ensure that all configured enforcement functions (authentication, access control rules, etc.) must be invoked prior to allowing a user to gain access to TOE or TOE mediated services. The TOE restricts the ability to modify the security attributes associated with access control rules, access to authenticated and unauthenticated services, etc to the Security Administrator. This feature ensures that no other user can modify the information flow policy to bypass the intended TOE security policy.
T.UNIDENTIFIED_ACTIONS: The administrator may not have the ability to notice potential security violations, thus limiting the administrator's ability to identify and take action against a possible security breach.	**O.AUDIT_REVIEW**: The TOE will provide the capability to selectively view audit information. **O.AUDIT_GENERATION** The TOE will provide the capability to detect and create records of security relevant events associated with users. **O.TIME_STAMPS** The TOE shall provide reliable time stamps for accountability and protocol purposes.	**O.AUDIT_REVIEW** helps to mitigate this threat by providing the Security Administrator with a required minimum set of configurable audit events that could indicate a potential security violation. By configuring these auditable events, the TOE monitors the occurrences of these events (e.g. set number of authentication failures, set number of information policy flow failures, self-test failures, etc.). **O.AUDIT_GENERATION** helps to mitigate this threat by recording actions for later review. **O.TIME_STAMPS** helps to mitigate this threat by ensuring that audit records have correct timestamps.
P.ACCESS_BANNER: The TOE shall display an initial banner describing restrictions of use, legal agreements, or any other appropriate information to which users consent by accessing the system.	**O.DISPLAY_BANNER**: The TOE will display an advisory warning regarding use of the TOE.	**O.DISPLAY_BANNER** satisfies this policy by ensuring that the TOE displays a Security Administrator configurable banner that provides all interactive users with a warning about the unauthorized use of the TOE.

Threat/Policy	Objectives Addressing the Threat and Policies	Rationale
P.ACCOUNTABILITY: The authorized users of the TOE shall be held accountable for their actions within the TOE.	**O.AUDIT_GENERATION**: The TOE will provide the capability to detect and create records of security-relevant events associated with users. **O.TIME_STAMPS**: The TOE shall provide reliable time stamps and the capability for the administrator to set the time used for these time stamps. **O.TOE_ACCESS**: The TOE will provide mechanisms that control a user's logical access to the TOE.	**O.AUDIT_GENERATION addresses** this policy by providing the Security Administrator with the capability of configuring the audit mechanism to record the actions of a specific user, or review the audit trail based on the identity of the user. Additionally, the administrator's ID is recorded when any security relevant change is made to the TOE (e.g. access rule modification, start-stop of the audit mechanism, establishment of a trusted channel, etc.). **O.TIME_STAMPS** plays a role in supporting this policy by requiring the TOE to provide a reliable time stamp (configured locally by the Security Administrator or via an external NTP server). The audit mechanism is required to include the current date and time in each audit record. All audit records that include the user ID, will also include the date and time that the event occurred. **O.TOE_ACCESS** supports this policy by requiring the TOE to identify and authenticate all authorized users prior to allowing any TOE access or any TOE mediated access on behalf of those users. While the user ID of authorized users can be assured, since they are authenticated, this PP allows unauthenticated users to access the TOE and the identity is then a presumed network identifier (e.g., IP address).
P.CRYPTOGRAPHY: Only NIST FIPS validated cryptography (methods and implementations) are acceptable for key management (i.e.; generation, access, distribution, destruction, handling, and storage of keys) and cryptographic services (i.e.; encryption, decryption, signature, hashing, key exchange, and random number generation services).	**O.CRYPTOGRAPHY**: The TOE shall use NIST FIPS 140-2 validated cryptographic services. **O.RESIDUAL_INFORMATION**: The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated.	**O.CRYPTOGRAPHY** satisfies this policy by requiring the TOE to implement NIST FIPS validated cryptographic services. These services will provide confidentiality and integrity protection of TSF data while in transit to remote parts of the TOE. O.RESIDUAL_INFORMATION satisfies this policy by ensuring that cryptographic data are cleared from resources that are shared between users. Keys must be zeroized according to FIPS 140-2 and the storage location for the keys must be overwritten three or more times upon the transfer of keys to another location

APPENDIX B MAPPING OF BASIC ROBUSTNESS OBJECTIVES TO REQUIREMENT COMPONENTS

(Back to TOC)

Sample rationale is provided below. The PP authors should examine various NIAP evaluated PPs for examples of rationale.

Objectives	Requirements Addressing the Objective	Rationale
O.ADMIN_GUIDANCE: The TOE will provide administrators with the necessary information for secure management.	ADO_DEL.1 ADO_IGS.1 ADO_ADM.1 AGD_USR.1 AVA_MSU.1	**ADO_DEL.1** ensures that the administrator is provided documentation that instructs them how to ensure the delivery of the TOE, in whole or in parts, has not been tampered with or corrupted during delivery. This requirement ensures the administrator has the ability to begin their TOE installation with a *clean* (e.g., malicious code has not been inserted once it has left the developer's control) version of the TOE, which is necessary for secure management of the TOE. **ADO_IGS.1** ensures the administrator has the information necessary to install the TOE in the evaluated configuration. Often times a vendor's product contains software that is not part of the TOE and has not been evaluated. The Installation, Generation and Startup (IGS) documentation ensures that once the administrator has followed the installation and configuration guidance the result is a TOE in a secure configuration. **AGD_ADM.1** mandates the developer provide the administrator with guidance on how to operate the TOE in a secure manner. This includes describing the interfaces the administrator uses in managing the TOE, security parameters that are configurable by the administrator, how to configure the TOE's rule set and the implications of any dependencies of individual rules. The documentation also provides a description of how to setup and review the auditing features of the TOE. **AGD_USR.1** is intended for non-administrative users, but could be used to provide guidance on security that is common to both administrators and non-administrators (e.g., password management guidelines). Since the non-administrative users of this TOE are limited to proxy users it is expected that the user guidance would discuss the secure use of proxies and how the single-use authentication mechanism is used. The use of the single-use authentication mechanism would not have to be repeated in the administrator's guide. **AVA_MSU.1** ensures that the guidance documentation is complete and consistent, and notes

Objectives	Requirements Addressing the Objective	Rationale
		all requirements for external security measures.
O.AUDIT_GENERATION: The TOE will provide the capability to detect and create records of security-relevant events associated with users	FAU_GEN.1-NIAP-0407 FAU_GEN.2-NIAP-0410 FIA_USB.1-NIAP-0415 FAU_SEL.1-NIAP-0407	**FAU_GEN.1-NIAP-0407** defines the set of events that the TOE must be capable of recording. This requirement ensures that the Administrator has the ability to audit any security relevant event that takes place in the TOE. This requirement also defines the information that must be contained in the audit record for each auditable event. This requirement also places a requirement on the level of detail that is recorded on any additional security functional requirements an ST author adds to this PP. **FAU_GEN.2-NIAP-0410** ensures that the audit records associate a user identity with the auditable event. In the case of authorized users, the association is accomplished with the userid. In all other cases, the association is based on the source network identifier, which is presumed to be the correct identity, but cannot be confirmed since these subjects are not authenticated. **FIA_USB.1-NIAP-0415** plays a role is satisfying this objective by requiring a binding of security attributes associated with users that are authenticated with the subjects that represent them in the TOE. This only applies to authorized users, since the identity of unauthenticated users cannot be confirmed. Therefore, the audit trail may not always have the proper identity of the subject that causes an audit record to be generated (e.g., presumed network address of an unauthenticated user may be a spoofed address). **FAU_SEL.1-NIAP-0407** allows the Security Administrator to configure which auditable events will be recorded in the audit trail. This provides the administrator with the flexibility in recording only those events that are deemed necessary by site policy, thus reducing the amount of resources consumed by the audit mechanism.
O.AUDIT_PROTECTION: The TOE will provide the capability to protect audit information.	FAU_SAR.2 FAU_STG.1-NIAP-0429 FAU_STG.3 FAU_STG.NIAP-0414-1-	**FAU_SAR.2** restricts the ability to read the audit trail to the Audit Administrator, thus preventing the disclosure of the audit data to any other user. However, the TOE is not expected to prevent the disclosure of audit data if it has been archived or saved in another form (e.g., moved or copied to an

Objectives	Requirements Addressing the Objective	Rationale
	NIAP-0429 FMT_MOF.1	saved in another form (e.g., moved or copied to an ordinary file). The **FAU_STG** family dictates how the audit trail is protected. **FAU_STG.1-NIAP-0429 restricts** the ability to delete audit records to the Security Administrator. **FAU_STG.3** requires the TOE to alert the administrator when the audit trail becomes full, and **FAU_STG.NIAP-0414-1-0429**, defines the actions that must be available to the administrator, as well as the action to be taken if there is no response. This helps to ensure that audit records are kept until the Security Administrator deems they are no longer necessary. This requirement also ensures that no one has the ability to modify audit records (e.g., edit any of the information contained in an audit record). This ensures the integrity of the audit trail is maintained. **FMT_MOF.1** restricts the capability to modify the behavior of the audit and alarm functions to the Security Administrator. While the Audit Administrator has the capability to choose how they will review the audit trail, they do not have the capability to select what events are audited. This requirement ensures that only the Security Administrator can turn audit on or off, this ensuring users actions are audited according to a site defined policy.
O.AUDIT_REVIEW: The TOE will provide the capability to selectively view audit information,.	FAU_SAR.1 FAU_SAR.3	**FAU_SAR.1** provides the Audit Administrator with the capability to read all the audit data contained in the audit trail. This requirement also mandates the audit information be presented in a manner that is suitable for the Audit Administrator to interpret the audit trail, which is subject to interpretation. It is expected that the audit information be presented in such a way that the Audit Administrator can examine an audit record and have the appropriate information (that required by FAU_GEN.2) presented together to facilitate the analysis of the audit review. FAU_SAR.3 complements FAU_SAR.1 by providing the Audit Administrator the flexibility to specify criteria that can be used to search or sort the audit records residing in the audit trail. FAU_SAR.3 requires the Audit Administrator be able to establish the audit review criteria based on a userid and source subject identity, so that the actions of a user can be readily identified and analyzed. The criteria also includes a destination subject identity so the Audit Administrator can determine what network traffic is destined for an individual machine. Allowing the Audit Administrator to perform searches or sort the audit records based on dates, times, subject identities, destination service identifier, or transport

Objectives	Requirements Addressing the Objective	Rationale
		layer protocol provides the capability to extract the network activity to what is pertinent at that time in order facilitate the Audit Administrator's review. Being able to search on the destination service identifier affords the Audit Administrator the opportunity to see what traffic is destined for a service (e.g., TCP port) or set of services regardless of where the traffic originated. It is important to note that the intent of sorting in this requirement is to allow the Audit Administrator the capability to organize or group the records associated with a given criteria. For example, if the Audit Administrator wanted to see what network traffic was destined for the set of TCP ports 1-1024, they would be able to have the audit data presented in such a way that all the traffic for TCP port 1 was grouped together, all the traffic for port 2 was grouped together and so on.
O.CONFIGURATION_IDENTIFICATION: The configuration of the TOE is fully identified in a manner that will allow implementation errors to be identified, corrected with the TOE being redistributed promptly.	ACM_CAP.2 ALC_FLR.2	**ACM_CAP.2** addresses this objective by requiring that that there be a unique reference for the TOE, and that the TOE is labeled with that reference. It also requires that there be a CM system in place, and that the configuration items that comprise the TOE by uniquely identified. This provides a clear identification of the composition of the TOE. **ALC_FLR.2** addresses this objective by requiring that there be a mechanism in place for identifying flaws subsequent to fielding, and for distributing those flaws to entities operating the system.
O.CORRECT_ TSF_OPERATION: The TOE will provide the capability to test the TSF to ensure the correct operation of the TSF at a customer's site.	FPT_TST_(EXP).1	**FPT_TST_(EXP).1** is necessary to ensure the correctness of the TSF configuration files and TSF data. FPT_TST_(EXP).2 is necessary to ensure the integrity of the TSF executable code. If TSF software is corrupted it is possible that the TSF would no longer be able to enforce the security policies. This also holds true for TSF data, if TSF data is corrupt the TOE may not correctly enforce its security policies. The FPT_TST_(EXP).1 functional requirement includes the critical nature and specific handling of the cryptographic related TSF data. Since the cryptographic TSF data has specific FIPS PUB requirements associated with them it is important to ensure that any fielded testing on the integrity of these data maintains the same level of scrutiny as specified in the FCS functional requirements.
O.CRYPTOGRAPHY_VALIDATED: The TOE will use NIST FIPS 140-2 validated cryptomodules for cryptographic services implementing		See Instruction 20 for a general discussion of cryptography

Objectives	Requirements Addressing the Objective	Rationale
NIST-approved security functions and random number generation services used by cryptographic functions.		
O.DISPLAY_BANNER: The TOE will display an advisory warning regarding use of the TOE.	FTA_TAB.1	**FTA_TAB.1** meets this objective by requiring the TOE display a Security Administrator defined banner before a user can establish an authenticated session. This banner is under complete control of the Security Administrator in which they specify any warnings regarding unauthorized use of the TOE and remove any product or version information if they desire.
O.DOCUMENTED_DESIGN: The design of the TOE is adequately and accurately documented.	ADV_FSP_(EXP).1 ADV_HLD_(EXP).1 ADV_RCR.1	**ADV_FSP_(EXP).1** requires that the interfaces to the TOE be documented and specified. **ADV_HLD_(EXP).1** requires that the high level design of the TOE be documented and specified and that said design be shown to correspond to the interfaces. **ADV_RCR.1** requires that there be a correspondence between adjacent layers of the design decomposition.
O.MANAGE: The TOE will provide all the functions and facilities necessary to support the administrators in their management of the security of the TOE, and restrict these functions and facilities from unauthorized use.	FMT_MOF.1 FMT_MSA.1 FMT_MSA.2 FMT_MSA.3-NIAP-0429 FMT_MTD.1 FMT_REV.1 FMT_SMR.1	**FMT_MOF.1** requires that the ability to use particular TOE capabilities be restricted to the Administrator. **FMT_MSA.1** requires that the ability to perform operations on security attributes be restricted to particular roles. **FMT_MSA.2** provides the Security Administrator the capability to manipulate the security attributes to facilitate the construction of the rule set. An example of this would be to group a set of service identifiers that are to have the same rule applied, rather than having to specify a separate rule for each service identifier. **FMT_MSA.3-NIAP-0429** requires that default values used for security attributes are restrictive, and that the Administrator has the ability to override those values. **FMT_MTD.1** requires that the ability to manipulate TOE content is restricted to Administrators and authorized Content Providers. **FMT_REV.1** restricts the ability to revoke attributes to the administrator. **FMT_SMR.1** defines the specific security roles to be supported.
O.MEDIATE:	FDP_ACC.1	The FDP requirements were chosen to define the

Objectives	Requirements Addressing the Objective	Rationale
The TOE must protect user data in accordance with its security policy.	FDP_ACF.1-NIAP-0460	policies, the subjects, objects, and operations for how and when mediation takes place in the TOE. FDP_ACC.1 defines that an Access Control policy that will be enforced on a list of subjects acting on the behalf of users attempting to gain access to a list of named objects. All the operations between subject and object covered are defined by the TOE's policy. The "subjects" are generally the TOE's "Agents." The "named objects" are the designated web based resources (web server, directories, files, or objects) that the TOE is protecting. FDP_ACF.1.-NIAP-0460 defines the Security Attribute used to provide Access Control to objects based on the following TOE's Access Control policy
O.PARTIAL_FUNCTIONAL_TESTING: The TOE will undergo some security functional testing that demonstrates the TSF satisfies some of its security functional requirements.	ATE_COV.1 ATE_FUN.1 ATE_IND.2	**ATE_FUN.1** requires that developer provide test documentation for the TOE, including test plans, test procedure descriptions, expected test results, and actual test results. These needs to identify the functions tested, the tests performed, and test scenarios. They require that the developer run those tests, and show that the expected results were achieved. **ATE_COV.1** requires that there be a correspondence between the tests in the test documentation and the TSF as described in the functional specification. **ATE_IND.2** requires that the evaluators test a subset of the TSF to confirm correct operation, on an equivalent set of resources to those used by the developer for testing. These sets should include a subset of the developer run tests.
O.RESIDUAL_ INFORMATION: The TOE will ensure that any information contained in a protected resource within its Scope of Control is not released when the resource is reallocated.	FDP_RIP.2	**FDP_RIP.2** is used to ensure the contents of resources are not available to subjects other than those explicitly granted access to the data. Also See Instruction 20 for a general discussion of cryptography
O.PARTIAL_SELF_PROTECTION The TSF will maintain a domain for its own execution that protects itself and its resources from external interference, tampering, or unauthorized disclosure through its own interfaces.	FPT_SEP_(EXP).1 FPT_RVM.1	The explicitly specific component **FPT_SEP_(EXP)**.1 was chosen to ensure the TSF provides a domain that protects itself from untrusted users. If the TSF cannot protect itself it cannot be relied upon to enforce its security policies. The explicitly specified version was used to distinguish the aspects of FPT_SEP provided by the TOE vs. the aspects provided by the IT environment. The inclusion of **FPT_RVM.1** ensures that the TSF makes policy decisions on all interfaces that perform operations on subjects and objects that are scoped by

Objectives	Requirements Addressing the Objective	Rationale
		the policies. Without this non-bypassability requirement, the TSF could not be relied upon to completely enforce the security policies, since an interface(s) may otherwise exist that would provide a user with access to TOE resources (including TSF data and executable code) regardless of the defined policies. This includes controlling the accessibility to interfaces, as well as what access control is provided within the interfaces.
O.TIME_STAMPS The TOE will provide reliable time stamps for accountability and protocol purposes.	FPT_STM.1	**FPT_STM.1** requires that the TSF provide time stamps for its own use.
O.TOE_ACCESS The TOE will provide mechanisms that control a user's logical access to the TOE.	FIA_AFL.1-NIAP-0425 FIA_ATD.1 FIA_UID FIA_UAU AVA_SOF AVA_SOF.1 FTA_SSL	**FIA_AFL.1-NIAP-0425** provides a detection mechanism for unsuccessful authentication attempts by remote administrators, authenticated proxy users and authorized IT entities. The requirement enables a Security Administrator settable threshold that prevents unauthorized users from gaining access to authorized user's account by guessing authentication data by locking the targeted account. Thus, limiting an unauthorized user's ability to gain unauthorized access to the TOE. **FIA_ATD.1** defines the attributes of users, including a userid that is used to by the TOE to determine a user's identity and enforce what type of access the user has to the TOE (e.g., the TOE associates a userid with any role(s) they may assume). **FIA_UID.1** requires that a user be identified to the TOE in order to access anything other than public content. **FIA_UAU.1** requires that a user be authenticated by the TOE before accessing anything other than public content. **FIA_UAU.7** provides that the authentication data provided by the user is not echoed back in plaintext, thus serving to protect that data. The **FTA_SSL** components all deal with automatic session locking and termination, either initiated by the TSF or a user The **AVA_SOF.1** requirement is applied to the password mechanism used by the local administrator (The single use authentication mechanism supplied by the IT environment (i.e., authentication server) has this same assurance requirement levied against it to ensure a consistent level of assurance.) For this TOE, the strength of function specified is medium. This requirement ensures the developer has

Objectives	Requirements Addressing the Objective	Rationale
		performed an analysis of the password mechanism to ensure the probability of guessing a local administrator's password would require a high-attack potential, as defined in Annex B of the CEM. This analysis takes into account the password space, as well as any feature of the password mechanism that plays a role in limiting the number of failed authentication attempts within a given time period.
O.VULNERABILITY_ANALYSIS: The TOE will undergo some vulnerability analysis to demonstrate the design and implementation of the TOE does not contain any obvious flaws.	AVA_VLA.1	The **AVA_VLA.1** component provides the necessary level of confidence that vulnerabilities do not exist in the TOE that could cause the security policies to be violated. AVA_VLA.1 requires the developer to perform a systematic search for potential vulnerabilities in all the TOE deliverables. For those vulnerabilities that are not eliminated, a rationale must be provided that describes why these vulnerabilities cannot be exploited by a threat agent with a moderate attack potential, which is in keeping with the desired assurance level of this TOE. As with the functional testing, a key element in this component is that an independent assessment of the completeness of the developer's analysis is made, and more importantly, an independent vulnerability analysis coupled with testing of the TOE is performed. This component provides the confidence that security flaws do not exist in the TOE that could be exploited by a threat agent of moderate (or lower) attack potential to violate the TOE's security policies.

APPENDIX C: SAMPLE PP MAPPING SPREADSHEET

(Back to TOC)

As mentioned in the main body of the this guidance, it is helpful to keep track of the mapping between the threats/policies in the PP, the objectives that contribute to the mitigation of each threat and implementation of each policy, and the specific requirements from each objective that apply to each threat or component. While the PPRB recommends that the PP authors make a working copy of Table 7 and update it while they are working on the PP, Table 7 takes up many pages and it is sometimes difficult to get an overall view of the mappings. The PPRB has found that a spreadsheet provides this condensed view and proved useful in writing consistent PP according to the Basic Robustness Consistency Manual. As noted in the main text of this guidance, the spreadsheet is nothing more than Table 7 without the notes column or all of the text associated with each threat and objective. Additionally, it is not expected that the spreadsheet be part of the PP; it is instead a tool for the PP authors to use or not, as they wish. An example spreadsheet that is associated with this consistency manual is provided below.

Threats/Policies	Objectives	Common Criteria Function and Security Requirements				
T.ACCIDENTAL_ADMIN_ERROR	O.ADMIN_GUIDEANCE	ADO_DEL.1	ADO_IGS.1	AGD_ADM.1	AGD_USR.1	AVA_MSU.1
T.ACCIDENTAL_AUDIT_COMPROMISE	O.AUDIT_PROTECTION	FMT_MOF.1	FAU_SAR.2	FAU_STG.1-NIAP-0429	FAU_STG.3	FAU_STG.NIAP-0429-1
	O.RESIDUAL_INFORMATION	FDP_RIP.1				
	O.SELF_PROTECTION	FPT_SEP	FPT_RVM			
T.ACCIDENTAL_CRYPTO_COMPROMISE	O.RESIDUAL_INFORMATION	FIPS-140-2				
T.MASQUERADE	O.TOE_ACCESS	FIA_AFL.1-NIAP-0425	FIA_ATD.1	FIA_UID	FIA_UAU	AVA_SOF
T.POOR_DESIGN	O.CONFIGURATION_IDENTIFICATION	ACM_CAP.2	ALC_FLR.2			
	O.RATINGS_MAINTENANCE	AMA_AMP.1	AMA_CAT.1	AMA_EVD.1	AMA_SIA.1	
	O.DOCUMENTED_DESIGN	ADV_FSP.1	ADV_HLD.1	ADV_RCR.1		
	O.VULNERABILITY_ANALYSIS	AVA_VLA.1				
T.POOR_IMPLEMENTATION	O.CONFIGURATION_IDENTIFICATION	ACM_CAP.2	ALC_FLR.2			
	O.PARTIAL_FUNCTIONAL_TESTING	ATE_COV.1	ATE_FUN.1	ATE_IND.2		
	O.VULNERABILITY_ANALYSIS	AVA_VLA.1				
T.POOR_TEST	O.DOCUMENTED_DESIGN	ADV_FSP.1	ADV_HLD.1	ADV_RCV.1		

Threats/Policies	Objectives	Common Criteria Function and Security Requirements				
	O.CORRECT_ TSF_OPERATION	FPT_TST_EXP				
	O.PARTIAL_FUNCTI ONAL_TESTING	ATE_COV.1	ATE_FUN.1	ATE_IND.2		
	O.VULNERABILITY_ ANALYSIS	AVA_VLA.1				
T.RESIDUAL_DATA	O.RESIDUAL_INFOR MATION	FDP_RIP.1				
T.TSF_COMPROMISE	O.RESIDUAL_INFOR MATION	FDP_RIP.2				
	O.PARTIAL_SELF_PR OTECTION	FPT_SEP	FPT_RVM			
	O.MANAGE	FMT_MTD.1	FMT_MSA.1	FMT_MOF.1		
T.UNATTENDED_SESSI ON	O.TOE_ACCESS	FTA_SSL.1	FTA_SSL.2	FTA_SSL.3	AVA_SOF.1	
T.UNAUTHORIZED_AC CESS	O.MEDIATE	FDP_*				
T.UNIDENTIFIED_ACTI ONS	O.AUDIT_REVIEW	FAU_SAR.1	FAU_SAR.3			
P.ACCESS_BANNER	O.DISPLAY_BANNER	FTA_TAB.1				
P.ACCOUNTABILITY	O.AUDIT_GENERATI ON	FAU_GEN.1-NIAP-0407	FAU_GEN.2-NIAP-0410	FIA_USB.1-NIAP-0415	FAU_SEL.1	
	O.TIME_STAMPS	FPT_STM.1	FMT_MTD.1			
	O.TOE_ACCESS	FIA_UID				
P.CRYPTOGRAPHY	O.CRYPTOGRAPHY	FIPS 140-2				
	O.RESIDUAL_INFOR MATION					

APPENDIX D: PROTECTION PROFILE COVER SHEET TEMPLATE

(Back to TOC)

An example cover sheet is provided below and should be used as a template by the author of the protection profile. The author shall replace the [Technology Area] with the technology area of the protection profile. In addition, the date and version number of the profile should also be included.

US Government Protection Profile

[Technology Area]

For

Basic Robustness Environments

Month dd, yyyy
Version x.x