

**U.S. Government**  
**Router**  
**Protection Profile**  
**For**  
**Medium Robustness Environments**



**Information Assurance Directorate**

**Version 1.1**  
**July 25, 2007**

**Protection Profile Title:**

- 1 U.S. Government Router Protection Profile for Medium Robustness Environments.

**Common Criteria Version:**

- 2 This PP was developed using Version 2.2 of the Common Criteria (CC) and applying the National Information Assurance Partnership (NIAP) interpretations and Precedence Decisions as appropriate that have been published by Trust Technology Assessment Program/ Common Criteria Evaluation Standard Scheme (TTAP/CCEVS) Management as of August 13, 2004.
- 3 This Protection Profile “*U.S. Government Protection Profile Router for Medium Robustness Environments*” (PP) was updated using Version 3.1 of the Common Criteria (CC).
- 4 Editor’s note: The purpose of this update was to bring the PP up to the new CC 3.1 standard without changing the authors’ original meaning or purpose of the documented requirements. The original PP was developed using version 2.x of the CC. The CC version 2.3 was the final version 2 update that included all international interpretations. CC version 3.1 used the final CC version 2.3 Security Functional Requirements (SFR)s as the new set of SFRs for version 3.1. Some minor changes were made to the SFRs in version 3.1, including moving a few SFRs to Security Assurance Requirements (SAR)s. There may be other minor differences between some SFRs in the version 2.3 PP and the new version 3.1 SFRs. These minor differences were not modified to ensure the author’s original intent was preserved.
- 5 The version 3.1 SARs were rewritten by the common criteria international community. The NIAP/CCEVS staff developed an assurance equivalence mapping between the version 2.3 and 3.1 SARs. The assurance equivalent version 3.1 SARs replaced the version 2.3 SARs in the PP.
- 6 Any issue that may arise when claiming compliance with this PP can be resolved using the observation report (OR) and observation decision (OD) process.
- 7 Further information, including the status and updates of this protection profile can be found on the CCEVS website: <http://www.niap-ccevs.org/cc-scheme/pp/>. Comments on this document should be directed to [ppcomments@missi.ncsc.mil](mailto:ppcomments@missi.ncsc.mil). The email should include the title of the document, the page, the section number, the paragraph number, and the detailed comment and recommendation.

## Table of Contents

<b>1</b>	<b>Introduction to the Protection Profile.....</b>	<b>5</b>
1.1	PP Identification.....	5
1.2	Overview of the Protection Profile .....	5
1.3	Conventions .....	6
1.4	Glossary of Terms.....	7
1.5	Document Organization .....	7
<b>2</b>	<b>TOE Description .....</b>	<b>8</b>
2.1	Product Type.....	8
2.2	TOE Definition .....	9
2.3	General TOE Functionality.....	9
2.4	TOE Operation Environment.....	12
<b>3</b>	<b>Security Environment.....</b>	<b>13</b>
3.1	Threats.....	13
3.2	Organizational Security Policies.....	18
3.3	Assumptions.....	20
<b>4</b>	<b>Security Objectives .....</b>	<b>20</b>
4.1	TOE Security Objectives .....	20
4.2	Environment Security Objectives (Non-IT).....	24
<b>5</b>	<b>IT Security Requirements .....</b>	<b>25</b>
5.1	TOE Security Functional Requirements .....	25
5.2	Security Requirements for the IT Environment.....	84
5.3	TOE Security Assurance Requirements.....	84
<b>6</b>	<b>Rationale .....</b>	<b>100</b>
6.1	Rationale for TOE Security Objectives .....	100
6.2	Rationale for the Security Objectives for the Environment.....	136
6.3	Rationale for TOE Security Requirements .....	138
6.4	Rationale for Assurance Requirements.....	185
6.5	Rationale for Satisfying all Dependencies.....	186
6.6	Rationale for Extended Requirements .....	186
<b>7</b>	<b>Appendices.....</b>	<b>190</b>
A	References.....	190
B	Glossary .....	192
C	Acronyms .....	198
D	Robustness Environment Characterization .....	201
E	Refinements .....	206
F	Statistical Random Number Generator Tests.....	213

## List of Tables and Figures

Table 1 Medium Robustness Security Objectives .....	20
Table 2 Medium Robustness Environmental Security Objectives .....	24
Table 3 Security Functional Requirements.....	25
Table 4 Auditable Events Table.....	31
Table 8 Assurance Requirements.....	84
Table 6 Rationale for TOE Security Objectives .....	100
Table 7 Rational for IT Environmental Objectives.....	136
Table 8 Rationale for TOE Security Requirements .....	138
Table 9 Functional Requirement Dependencies .....	186
Table 10 Rationale for Extended Requirements .....	187

# 1 INTRODUCTION TO THE PROTECTION PROFILE

---

## 1.1 PP Identification

- 8 Title: U.S. Government Router PP for Medium Robustness Environment
- 9 Sponsor: National Security Agency (NSA)
- 10 CC Version: Common Criteria (CC) Version 2.2, and applicable interpretations, updated to CC version 3.1.
- 11 Registration: <to be provided upon registration>
- 12 PP Version: Version 1.1 dated July 25, 2007.
- 13 Keywords: Router, protection profile, encryption, decryption, IPSEC ESP, IPSEC AH, IKE,

## 1.2 Overview of the Protection Profile

- 14 The U.S. Government Router PP for Medium Robustness Environments specifies a set of security functional and assurance requirements for Information Technology (IT) products. A router monitors, routes and manipulates network traffic to facilitate its delivery for the proper destination on a network or between networks.
- 15 The Router PP is applicable to products regardless of whether they are externally or internally facing a given network. In addition, it addresses only security requirements and not any special considerations of any particular product design.
- 16 The Router PP was constructed to provide a target metric for the deployment of router devices. This protection profile identifies security functions and assurances that represent the lowest common set of requirements that must be addressed at a Medium Robustness level by a router.
- 17 STs that claim conformance to this PP shall meet a minimum standard of demonstrable-PP conformance as defined in section D3 of part 1.
- 18 The assurance requirements are presented in Section 5.3.
- 19 This PP defines:

- assumptions about the security aspects of the environment in which the TOE will be used;
- threats that are to be addressed by the TOE;
- security objectives of the TOE and its environment;
- functional and assurance requirements to meet those security objectives; and;
- rationale demonstrating how the requirements meet the security objectives, and how the security objectives address the threats.

### 1.3 Conventions

- 20 Except for replacing United Kingdom spelling with American spelling, the notation, formatting, and conventions used in this PP are consistent with version 2.2 of the CC. Words may be modified for readability by crossing out the original text and replacing it with the modified text. Selected presentation choices are discussed here to aid the PP reader.
- 21 The CC allows several operations to be performed on functional requirements; *refinement*, *selection*, *assignment*, and *iteration* are defined in paragraph 4.4.1.3.2 of Part 2 of the CC. Each of these operations is used in this PP.
- 22 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**, any text that is removed by the refinement will be crossed out (ie. ~~crossed out text~~). In some cases, End Notes were used when refinements delete text from the original requirement.
- 23 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 Security Target (ST) author appear in square brackets with an indication that a selection is to be made, [selection:], and are not italicized.
- 24 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:].

- 25 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).
- 26 As this PP was sponsored, in part by NSA, NIAP, interpretations are used and are presented with the NIAP interpretation number as part of the requirement identifier (e.g., **FAU\_GEN.1-NIAP-0429** for Audit data generation).
- 27 The PP authored reviewed the International Interpretations database and applied those interpretations as necessary
- 28 The CC paradigm also allows protection profile and security target authors to create their own requirements. Such requirements are termed “extended requirements” and are permitted if the CC does not offer suitable requirements to meet the authors’ needs. **Extended requirements** must be identified and are required to use the CC class/family/component model in articulating the requirements. In this PP, extended requirements will be indicated with the “(EXP)” following the component name.
- 29 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.

## 1.4 Glossary of Terms

- 30 See Appendix B for the Glossary.

## 1.5 Document Organization

- 31 Section 1, Introduction to the Protection Profile, provides the document management and overview information necessary to identify the PP.
- 32 Section 2, Target of Evaluation (TOE) Description, defines the TOE and establishes the context of the TOE by referencing generalized security functions.
- 33 Section 3, Security Environment, describes the expected environment in which the TOE is to be used. This section defines the set of threats that are relevant to the secure operation of the TOE, organizational security policies with which the TOE must comply, and secure usage assumptions applicable to this analysis.
- 34 Section 4, Security Objectives, defines the set of security objectives to be satisfied by the TOE and by the TOE operating environment.

- 35 Section 5, IT Security Requirements, specifies the security functional and assurance requirements that must be satisfied by the TOE and the IT environment.
- 36 Section 6, Rationale, provides rationale to demonstrate that the security objectives satisfy the threats and policies. This section also explains how the set of requirements are complete relative to the security objectives and presents a set of arguments that address dependency analysis and use of the extended requirement.
- 37 Section 7, Appendices, includes the appendices that accompany the PP and provides clarity and/or explanation for the reader.
- 38 Appendix A, References, provides background material for further investigation by users of the PP.
- 39 Appendix B, Glossary, provides a listing of definitions of terms.
- 40 Appendix C, Acronyms, provides a listing of acronyms used throughout the document.
- 41 Appendix D, Robustness Environment Characterization, contains a discussion characterizing the level of robustness TOEs compliant with the PP can achieve. The PPRB created a discussion that provides a definition of factors for TOE environments and an explanation of how a given level of robustness is categorized.
- 42 Appendix E, Refinements, identifies the refinements that were made to CC requirements where text is deleted from a requirement.
- 43 Appendix F, Statistical Random Number generator Tests.

## **2 TOE DESCRIPTION**

---

- 44 This Protection Profile specifies the minimum-security requirements to satisfy Medium Robustness Environments for a TOE that is a router.

### **2.1 Product Type**

- 45 Router PP-conformant products support the ability to monitor, route, and manipulate network traffic to facilitate its delivery to the proper destination on a network or between networks.

- 46 The Router PP was constructed to provide a target and metric for the deployment of router devices. This protection profile identifies security functions and assurances that represent the minimum set of security requirements that should be addressed at a Medium Robustness level by a router.
- 47 The Router PP is applicable to products regardless of whether they are externally or internally facing a given network. In addition, it addresses only security requirements and not any special considerations of any particular product design.
- 48 The Router PP addresses only those factors that should be considered when dealing with a dedicated router. It does not cover extra functionality that may be added to a router (such as point-to-point user network data encryption and detailed traffic monitoring and manipulation) that in essence changes the router to another type of device such as a Virtual Private Network (VPN) endpoint or a firewall. Those devices should be covered under their own appropriate protection profile document.

## 2.2 TOE Definition

- 49 A router is a device that forwards data packets along networks. A router is connected to at least two networks, commonly two LANs or WANs or a LAN and its ISP's network. Routers are located at gateways, the places where two or more networks connect. Routers use headers and forwarding tables to determine the best path for forwarding the packets, and they use protocols such as ICMP (Internet Control Message Protocol) to communicate with each other and configure the best route between any two hosts. A router can be configured to support multiple network protocols and routing protocols on a single device.

## 2.3 General TOE Functionality

- 50 A router that is compliant with the Router PP provides the following security functions in its evaluated configuration:
- Administration-“Administrators” refers to the roles assigned to the individuals responsible for the installation, configuration, and maintenance of the TOE. The TOE requires three separate administrative roles: Cryptographic Administrator, Audit Administrator and Security Administrator. The Cryptographic Administrator is responsible for the configuration and maintenance of cryptographic elements related to the establishment of secure connections to and from the TOE. The Audit Administrator is responsible for the regular review of the TOE's audit data. The Security Administrator is responsible for all other administrative tasks

(e.g., creating the TOE security policy) not addressed by the other two administrative roles. It is important to note that while this PP requires the three administrative roles outlined above, it provides the ST author the option of including additional administrative roles as well.

- Audit – Section 5.1.1 “Security Audit (FAU)” describes the TOE’s generation of auditable events, audit records, alarms and audit management. Table 4 in the FAU\_GEN.1-NIAP-0429 requirement lists the minimum set of auditable events that must be available to the Security Administrator for configuration on the TOE. Each auditable event must generate an audit record. Table 4 also provides a minimum list of attributes that must be included in each audit record. The ST author may include additional auditable events and audit record attributes. If the ST author includes any additional functional requirements not specified by this PP, they must consider any security relevant events associated with those requirements and include them in the TOE’s list of auditable events and records. In addition to generating auditable events, the TOE must monitor their occurrences and provide a Security Administrator configurable threshold for determining a potential security violation. Once the TOE has detected a potential security violation, an alarm is generated and a message is displayed at the TOE’s local console and each active remote administrator console (all administrative roles included). Additionally, the Security Administrator can configure the TOE to generate an audible alarm to indicate a potential security violation. If an administrator console is not active, the TOE stores the message for display when the console becomes active (e.g., when the administrator establishes a remote session to the TOE). The message must contain the potential security violation and all audit records associated with the potential security violation. The message will be displayed at the various consoles until administrator acknowledgement of the message has occurred. As mentioned in the “Administrative” section above, the Audit Administrator’s role is restricted to viewing the contents of the audit records and the deletion of the audit trail. The TOE does provide the Audit Administrator with a sorting and searching capability to improve audit analysis. The Security Administrator configures auditable events, backs-up and manages audit data storage. The TOE provides the Security Administrator with a configurable audit trail threshold to track the storage capacity of the audit trail. Once the threshold is met, the TOE generates an alarm and displays a message in the same fashion as described above, including the option of the audible alarm. In addition to displaying the message, the Security Administrator may configure the TOE to prevent all auditable events except for those performed by the Security and Audit Administrators or overwrite the oldest audit records in the audit trail.

- Cryptography – Cryptographic algorithms and key management functions that meet published standards are required in Router PP-complaint products. Section 5.1.2 “Cryptographic Support” defines the minimum set of cryptographic attributes required by the TOE. The TOE’s cryptographic module(s) must be FIPS PUB 140-2 validated. The ST author may implement the cryptographic module(s) in hardware, software, or a combination of both. The TOE must generate and distribute symmetric and asymmetric keys. The ST author is provided several implementation selections for key generation and may distribute keys manually, electronically, or both. The TOE must perform data encryption/decryption using the Triple Data Encryption Algorithm (TDEA) with a minimum key size of 168 bits or the Advanced Encryption Standard (AES) with a minimum key size of 192 bits. Additional requirements for key destruction, digital signature generation/verification, random number generation and cryptographic hashing and message authentication are provided in section 5.1.2.
- Identification and Authentication – The TOE requires multiple Identification and Authentication (I&A) mechanisms for access to services residing on the TOE or for services mediated by the TOE. The type of authentication mechanism required depends on the origin of the source (i.e., remote user, TOE console) requesting the service.
- Trusted Channel/Trusted Path- Trusted paths must be established between the router and the management station and (trusted) channels must be established between individual routers in order to exchange management information. Between routers, network control information is exchanged via (trusted) channels to allow dynamic connection establishment and packet routing. Network control information consists of specific requests and instructions that include destination address, routing controls, and signalling information. Examples of control information in the IP environment include Open Shortest Path First (OSPF), BGP, Resource Reservation Protocol (RSVP), and Label Distribution Protocol (LDP). The TOE is required to provide two types of encrypted communications: trusted channel and trusted path. Trusted channel refers to the encrypted connection between the TOE and a trusted IT entity. Trusted path refers to the encrypted connection used to authenticate an administrator with the TOE.
- Additional Security Functionality – The TOE is also required to provide:
  - User data protection, including information flow control and related flow control policies, definition of security attributes;

- TOE session establishment; The intent of this requirement is to ensure that only valid administrative session or a authenticated user's are valid and permitted to use the router resources.
- Resource utilization functions; The TSF will protects itself and its resources from external interference, tampering or unauthorized disclosure
- Protection of the TSF; The TSF will ensure that security services are not bypassed or tampered with. The TSF and the OS cooperatively provide this service.

## 2.4 TOE Operation Environment

- 51 A router is placed at the edge of a given network or network segment. For a router to function it must have at least two distinct networks or network segments to pass data between. Router data refers to all TOE Security Functions (TSF) data dealing with the functionality of the Router (e.g., Router audit data, signatures, policies, etc.). User data is that data that flows between authenticated or unauthenticated networks that do not affect the operations of the TSF.

### 3 SECURITY ENVIRONMENT

---

- 52 A medium robustness TOE is considered sufficient protection for environments where the likelihood of an attempted compromise is medium. This implies that the motivation of the threat agents will be average in environments that are suitable for TOEs of medium robustness. Note that this also implies that the resources and expertise of the threat agents really are not factors that need to be considered, because highly sophisticated threat agents will not be motivated to use great expertise or extensive resources in an environment where medium robustness is suitable.
- 53 The medium motivation of the threat agents can be reflected in a variety of ways. One possibility is that the value of the data processed or protected by the TOE will be only medium, thus providing little motivation of even a totally unauthorized entity to attempt to compromise the data. Another possibility, (where higher value data is processed or protected by the TOE) is that the procuring organization will provide environmental controls (that is, controls that the TOE itself does not enforce) in order to ensure that threat agents that have generally high motivation levels (because of the value of the data) cannot logically or physically access the TOE (e.g., all users are “vetted” to help ensure their trustworthiness, and connectivity to the TOE is restricted).
- 54 The remainder of this section addresses the following:
- Threats to TOE assets or to the TOE environment which must be countered;
  - Organizational Security Policies that compliant TOEs must enforce; and
  - Assumptions about the security aspects of a compliant TOE environment.
- 55 It is important to note to vendors and end users that any IT entity that is used to protect National Security information, and employs cryptography as a protection mechanism, will require the TOE’s key management techniques to be approved by NSA prior to the fielding of the TOE.

#### 3.1 Threats

##### 3.1.1 Threat Agent Characterization

- 56 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).

- 57 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.
- 58 Unlike the motivation factor, however, the same cannot 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.
- 59 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*.
- 60 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).
- 61 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.

- 62 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.
- 63 The important general points 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 are “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.
- 64 The following threats are addressed by the TOE and should be read in conjunction with the threat rationale, Section 6.1. There are other threats that the TOE does not address (e.g., malicious developer inserting a backdoor into the TOE) and it is up to a site to determine how these types of threats apply to its environment.

**Table 1 Medium Robustness Applicable Threats**

Threat Name	Threat Definition
T.ADMIN_ERROR	An administrator may incorrectly install or configure the TOE, or install a corrupted TOE resulting in ineffective security mechanisms.
T.ADMIN_ROGUE	An administrator's intentions may become malicious resulting in user or TOE Security Functions (TSF) data being compromised.
T.AUDIT_COMPROMISE	A malicious 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.
T.CRYPTO_COMPROMISE	A malicious 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.
T.FLAWED_DESIGN	Unintentional or intentional errors in requirements specification or design of the TOE may occur, leading to flaws that may be exploited by a malicious user or program.
T.FLAWED_IMPLEMENTATION	Unintentional or intentional errors in implementation of the TOE design may occur, leading to flaws that may be exploited by a malicious user or program.
T.MALICIOUS_TSF_COMPROMISE	A malicious user or process may cause TSF data or executable code to be inappropriately accessed (viewed, modified, or deleted).

Threat Name	Threat Definition
T.MASQUERADE	A malicious user, process, or external IT entity may masquerade as an authorized entity in order to gain access to data or TOE resources.
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.
T.REPLAY	A user may gain inappropriate access to the TOE by replaying authentication information, or may cause the TOE to be inappropriately configured by replaying TSF data or security attributes (e.g., captured as transmitted during the course of legitimate use).
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.
T.RESOURCE_EXHAUSTION	A malicious process or user may block others from system resources (e.g., <i>connection state tables, TCP connections</i> ) via a resource exhaustion denial of service attack.
T.SPOOFING	A malicious user, process, or external IT entity may misrepresent itself as the TOE to obtain identification and authentication data.
T.UNATTENDED_SESSION	A user may gain unauthorized access to an unattended session.
T.UNAUTHORIZED_ACCESS	A user may gain access to user data for which they are not authorized according to the TOE security policy.

Threat Name	Threat Definition
T.UNIDENTIFIED_ACTIONS	The administrator may fail to notice potential security violations, thus limiting the administrator's ability to identify and take action against a possible security breach.
T.UNAUTHORIZED_PEER	An unauthorized IT entity may attempt to establish a security association with the TOE.
T.UNKNOWN_STATE	When the TOE is initially started or restarted after a failure, the security state of the TOE may be unknown.
T.EAVESDROP	A malicious user or process may observe or modify user or TSF data transmitted between physically separated parts of the TOE.

### 3.2 Organizational Security Policies

- 65 An organizational security policy is a set of rules, practices, and procedures imposed by an organization to address its security needs.

**Table 2 Medium Robustness Applicable Policies**

<b>Policy Name</b>	<b>Policy Definition</b>
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 TOE.
P.ACCOUNTABILITY	The authorized users of the TOE shall be held accountable for their actions within the TOE.
P.ADMIN_ACCESS	Administrators shall be able to administer the TOE both locally and remotely through protected communications channels.
P.CRYPTOGRAPHY	The TOE shall use NIST FIPS validated cryptography as a baseline with additional NSA-approved methods for key management (i.e.; generation, access, distribution, destruction, handling, and storage of keys), and for cryptographic operations (i.e.; encryption, decryption, signature, hashing, key exchange, and random number generation services).
P.VULNERABILITY_ANALYSIS_TEST	The TOE must undergo appropriate independent vulnerability analysis and penetration testing to demonstrate that the TOE is resistant to an attacker possessing a medium attack potential.
P.COMPATIBILITY	The TOE must meet Request for Comments (RFC) requirements for implemented protocols to facilitate inter-operation with other routers and network equipment using the same protocols.

### 3.3 Assumptions

- 66 This section contains assumptions regarding the security environment and the intended usage of the TOE.

**Table 3 Medium Robustness Applicable Assumptions**

Assumption Name	Assumption Definition
A.NO_GENERAL_PURPOSE	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.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.
A.AVAILABILITY	Network resources shall be available to allow clients to satisfy mission requirements and to transmit information.

## 4 SECURITY OBJECTIVES

---

- 67 This section identifies the security objectives of the TOE and its supporting environment. The security objectives identify the responsibilities of the TOE and its environment in meeting the security needs.

### 4.1 TOE Security Objectives

**Table 1 Medium Robustness Security Objectives**

Objective Name	Objective Definition
O.ADMIN_ROLE	The TOE will provide administrator roles to isolate administrative actions, and to make the administrative functions available locally and remotely.

Objective Name	Objective Definition
O.AUDIT_GENERATION	The TOE will provide the capability to detect and create records of security-relevant events associated with users.
O.AUDIT_PROTECTION	The TOE will provide the capability to protect audit information.
O.AUDIT_REVIEW	The TOE will provide the capability to selectively view audit information, and alert the administrator of identified potential security violations.
O.CHANGE_MANAGEMENT	The configuration of, and all changes to, the TOE and its development evidence will be analyzed, tracked, and controlled throughout the TOE's development.
O.CORRECT_TSF_OPERATION	The TOE will provide a capability to test the TSF to ensure the correct operation of the TSF in its operational environment.
O.CRYPTOGRAPHIC_FUNCTIONS	The TOE shall provide cryptographic functions (i.e., encryption/decryption and digital signature operations) to maintain the confidentiality and allow for detection of modification of TSF data that is transmitted between physically separated portions of the TOE, or stored outside the TOE.
O.DISPLAY_BANNER	The TOE will display an advisory warning regarding use of the TOE.
O.DOCUMENT_KEY_LEAKAGE	The bandwidth of channels that can be used to compromise key material shall be documented.
O.MAINT_MODE	The TOE shall provide a mode from which recovery or initial startup procedures can be performed.

Objective Name	Objective Definition
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.MEDIATE_INFORMATION_FLOW	The TOE must mediate the flow of information between sets of TOE network interfaces or between a network interface and the TOE itself in accordance with its security policy.
O.PEER_AUTHENTICATION	The TOE will authenticate each peer TOE that attempts to establish a security association with the TOE.
O.PROTOCOLS	The TOE will ensure that standardized protocols are implemented in the TOE to RFC and/or Industry specifications to ensure interoperability.
O.PROTECT_IN_TRANSIT	The TSF shall protect TSF data when it is in transit between the TSF and another trusted IT entity.
O.REPLAY_DETECTION	The TOE will provide a means to detect and reject the replay of authentication data and other TSF data and security attributes.
O.RESIDUAL_INFORMATION	The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated.
O.RESOURCE_SHARING	The TOE shall provide mechanisms that mitigate attempts to exhaust connection-oriented resources provided by the TOE (e.g., entries in a connection state table; TCP connections to the TOE).

Objective Name	Objective Definition
O. ROBUST_ADMIN_GUIDANCE	The TOE will provide administrators with the necessary information for secure delivery and management.
O.ROBUST_TOE_ACCESS	The TOE will provide mechanisms that control a user's logical access to the TOE and to explicitly deny access to specific users when appropriate.
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.
O.SOUND_DESIGN	The TOE will be designed using sound design principles and techniques. The TOE design, design principles and design techniques will be adequately and accurately documented.
O.SOUND_IMPLEMENTATION	The implementation of the TOE will be an accurate instantiation of its design, and is adequately and accurately documented.
O.THOROUGH_FUNCTIONAL_TESTING	The TOE will undergo appropriate security functional testing that demonstrates the TSF satisfies the security functional requirements.
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.TRUSTED_PATH	The TOE will provide a means to ensure that users are not communicating with some other entity pretending to be the TOE when supplying identification and authentication data.
O.USER_GUIDANCE	The TOE will provide users with the information necessary to correctly use the security mechanisms.

Objective Name	Objective Definition
O.VULNERABILITY_ANALYSIS_TEST	The TOE will undergo appropriate independent vulnerability analysis and penetration testing to demonstrate the design and implementation of the TOE and does not allow attackers with medium attack potential to violate the TOE's security policies.

## 4.2 Environment Security Objectives (Non-IT)

**Table 2 Medium Robustness Environmental Security Objectives**

Environmental Objective Name	Environmental Objective Definition
OE.NO_GENERAL_PURPOSE	The Administrator ensures there are no general-purpose computing or storage repository capabilities (e.g., compilers, editors, or user applications) available on the TOE.
OE.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the IT environment.
OE.AVAILABILITY	Network resources will be available to allow clients to satisfy mission requirements and to transmit information.

## 5 IT SECURITY REQUIREMENTS

---

### 5.1 TOE Security Functional Requirements

- 68 This section defines the functional requirements for the TOE. Functional requirements in this PP were drawn directly from Part 2 of the CC, or were based on Part 2 of the CC. These requirements are relevant to supporting the secure operation of the TOE.

**Table 3 Security Functional Requirements**

<b>Functional Components (from CC Part 2)</b>	
FAU_ARP.1	Security alarms
FAU_ARP_ACK_(EXT).1	Security alarm acknowledgement
FAU_GEN.1-NIAP-0429	Audit data generation
FAU-GEN.2-NIAP-0410	User identity association
FAU_SAA.1-NIAP-0407	Potential violation analysis
FAU_SAR.1	Audit review
FAU_SAR.2	Restricted audit review
FAU_SAR.3	Selectable audit review
FAU_SEL.1-NIAP-0407	Audit event selection
FAU_STG.NIAP-0414-1-NIAP-0429	Site-configurable prevention of audit loss
FAU_STG.1-NIAP-0429	Audit event storage
FAU_STG.3	Action in case of possible audit data loss
FCS_BCM_(EXT).1	Baseline Cryptographic Module
FCS_CKM.1(1)	Cryptographic key generation (for symmetric keys using Random Number Generator (RNG))

<b>Functional Components (from CC Part 2)</b>	
FCS_CKM.1(2)	Cryptographic key generation (for asymmetric keys)
FCS_CKM.2	Cryptographic key distribution
FCS_CKM.4	Cryptographic key destruction
FCS_CKM_(EXT).2	Cryptographic Key Handling and Storage
FCS_COP.1(1)	Cryptographic operation (for data encryption/decryption)
FCS_COP.1(2)	Cryptographic operation (for cryptographic signature)
FCS_COP.1(3)	Cryptographic operation (for cryptographic hashing)
FCS_COP.1(4)	Cryptographic operation (for cryptographic key agreement)
FCS_COP_(EXT).1	Random Number Generation
FCS_IKE_(EXT).1	Internet Key Exchange
FDP_IFC.1(1)	Subset information flow control (unauthenticated TOE services policy)
FDP_IFC.1(2)	Subset information flow control (authenticated TOE services policy)
FDP_IFF.1(1)	Simple security attributes (unauthenticated policy)
FDP_IFF.1(2)	Simple security attributes (authenticated policy)
FDP_RIP.2	Full residual information protection
FIA_AFL.1	Authentication failure handling
FIA_ATD.1(1)	User attribute definition (Human users)
FIA_ATD.1(2)	User attribute definition (TOE to TOE Identification)
FIA_UAU.2	User authentication before any action
FIA_UAU_(EXT).5	Authentication Mechanism

<b>Functional Components (from CC Part 2)</b>	
FIA_UID.2	User identification before any action
FIA_USB.1	User-Subject Binding
FMT_MOF.1(1)	Management of security functions behavior (TSF non-cryptographic self-test)
FMT_MOF.1(2)	Management of security functions behavior (cryptographic self-test)
FMT_MOF.1(3)	Management of security functions behavior (Audit Review)
FMT_MOF.1(4)	Management of security functions behavior (Audit Selection)
FMT_MOF.1(5)	Management of security functions behavior (Alarms)
FMT_MOF.1(6)	Management of security functions behavior (quota mechanism)
FMT_MOF.1(7)	Management of security functions behavior (unsuccessful authentication attempts).
FMT_MSA.1(1)	Management of security attributes (unauthenticated)
FMT_MSA.1(2)	Management of security attributes (authenticated)
FMT_MSA.2	Secure security attributes
FMT_MSA.3(1)	Static attribute initialization (unauthenticated services)
FMT_MSA.3(2)	Static attribute initialization (authenticate services)
FMT_MTD.1(1)	Management of TSF data (non-cryptographic, non-time TSF data)
FMT_MTD.1(2)	Management of TSF data (cryptographic TSF data)
FMT_MTD.1(3)	Management of TSF data (time TSF data)
FMT_MTD.1(4)	Management of TSF data (Router Policy Ruleset)

<b>Functional Components (from CC Part 2)</b>	
FMT_MTD.2(1)	Management of limits on TSF data (transport-layer quotas)
FMT_MTD.2(2)	Management of limits on TSF data (controlled connection-oriented quotas)
FMT_MTD.2(2)	Management of limits on TSF data (percentage of storage capacity for audit records)
FMT_REV.1	Revocation
FMT_SMF.1	Specification of Management Functions
FMT_SMR.2	Restrictions on security roles
FPT_FLS.1	Failure with preservation of secure state
FPT_ITA.1	Inter-TSF availability within a defined availability metric
FPT_ITC.1	Inter-TSF confidentiality during transmission
FPT_ITI.1	Inter-TSF detection of modification
FPT_PRO_(EXT).1	Standard protocol usage
FPT_RCV.2	Automated Recovery
FPT_RPL.1	Replay detection
FPT_STM.1	Reliable time stamps
FPT_TDC.1	Inter-TSF basic TSF data consistency
FPT_TST_(EXT).1	TSF testing
FPT_TST.1(1)	TSF testing (Cryptographic)
FPT_TST.1(2)	TSF testing (key generation)
FRU_RSA.1(1)	Maximum quotas (transport-layer quotas)
FRU_RSA.1(2)	Maximum quotas (controlled connection-oriented quotas)

<b>Functional Components (from CC Part 2)</b>	
FTA_SSL.3	TSF-initiated termination
FTA_TAB.1	Default TOE access banners
FTA_TSE.1	TOE session establishment
FTP_ITC.1(1)	Inter-TSF trusted channel (Prevention of Disclosure)
FTP_ITC.1(2)	Inter-TSF trusted channel (Detection of Modification)
FTP_TRP.1(1)	Trusted path (Prevention of Disclosure)
FTP_TRP.1(2)	Trusted path (Detection of Modification)

### 5.1.1 Security Audit (FAU)

#### 5.1.1.1 Security alarms (FAU\_ARP.1)

FAU\_ARP.1.1 The TSF shall ~~take~~ [immediately display a message identifying the potential security violation, and make accessible the audit record contents associated with the auditable event(s) that generated the alarm, at the

- a) local console;
- b) remote Security Administrator sessions that exist;
- c) remote Security Administrator sessions that are initiated before the alarm has been acknowledged; and
- d) [selection: [ST assignment: other methods determined by the ST author], no other methods]

] upon detection of a potential security violation.

- 69 Application Note: The TSF provides a message to the local console regardless of whether an administrator is logged in. The message is displayed at the remote console if an administrator is already logged in, or when an administrator logs in if the alarm message has not been acknowledged. The audit records contents associated with the alarm (See FAU\_ARP\_ACK\_(EXT).1) may or may not be part of the message displayed; however the relevant audit information must be available to administrators. In addition, the TOE provides an audible alarm

(FAU\_ARP\_ACK\_(EXT).1) that can be configured to sound an alarm if desired by the Security Administrator. It is acceptable for the ST author to fill the open assignment without any, if no other methods (e.g., pager, e-mail) are included in the TOE.

#### 5.1.1.2 Security alarm acknowledgement (FAU\_ARP\_ACK\_ (EXP).1)

FAU\_ARP\_ACK\_(EXT).1.1 – The TSF shall display the alarm message identifying the potential security violation and make accessible the audit record contents associated with the auditable event(s) until it has been acknowledged. If the Security Administrator configures the TOE to generate an optional audible alarm, the audible alarm will sound until acknowledged by an administrator. Once the alarm is acknowledged, it will be reset to zero.

FAU\_ARP\_ACK\_(EXT).1.2 – The TSF shall display an acknowledgement message identifying a reference to the potential security violation, a notice that it has been acknowledged, the time of the acknowledgement and the user identifier that acknowledged the alarm, at the:

- local console, and
- remote administrator sessions that received the alarm.

70 Application Note: This extended requirement is necessary since a CC requirement does not exist to ensure an administrator will be aware of the alarm. The intent is to ensure that if an administrator is logged in and not physically at the console or remote workstation the message will remain displayed until they have acknowledged it. The message will not be scrolled off the screen due to other activity-taking place (e.g., the Audit Administrator is running an audit report). If the Security Administrator configures the TOE to generate an audible alarm, the alarm will sound until an administrator acknowledges the alarm. Acknowledging the message and audible alarm could be a single event, or different events.

71 FAU\_ARP\_ACK\_(EXT).1.2 ensures that each administrator that received the alarm message also receives the acknowledgement message, which includes some form of reference to the alarm message, who acknowledged the message and when.

#### 5.1.1.3 Audit data generation ( FAU\_GEN.1-NIAP-0429)

FAU\_GEN.1.1-NIAP-04070429 The TSF shall be able to generate an audit record of the following auditable events:

- a) Start-up and shutdown of the audit functions;
- b) All auditable events for the *basic* level of audit; and

c) [specifically defined auditable events listed in Table 4, [events introduced by the inclusion of additional SFRs determined by the ST author], [events introduced by the inclusion of extended requirements determined by the ST author], “no additional events”].

72 *Application Note: For the first assignment in the selection, 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.*

73 *Likewise, for the second assignment the ST author includes audit events that may arise due to the inclusion of any extended 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. It is acceptable for the ST author to choose “no additional events”, if the ST author has not included additional requirements, or has included additional requirements that do not have a basic level (or commensurate level) of audit associated with them.*

FAU\_GEN.1.2-~~NIAP-04070429~~ The TSF shall record within each audit record at least the following information:

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

b) For each audit event time, based on the auditable event definitions of the functional components included in the PP/ST, [information specified in column three of Table 4 below]

**Table 4 Auditable Events Table**

<b>Requirement</b>	<b>Auditable Events</b>	<b>Audit Record Contents</b>
FAU_ARP.1	Actions taken due to potential security violations.	Identification of what caused the generation of the alarm.
FAU_ARP_ACK_(EXT).1	Actions taken due to potential security violations.	The identity and location of the administrator that acknowledged the alarm.
FAU_GEN.1-NIAP-0429	None.	

<b>Requirement</b>	<b>Auditable Events</b>	<b>Audit Record Contents</b>
FAU_GEN.2-NIAP-0410	None.	
FAU_SAA.1-NIAP-0407	Enabling and disabling of any of the analysis mechanisms;  Automated responses performed by the tool.	The identity of the Security Administrator performing the function.
FAU_SAR.1	Reading of information from the audit records.	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.
FAU_SAR.3	Unsuccessful attempts to read information from the audit records.	The parameters used for the viewing.
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.
FAU_STG.NIAP-0414-1-NIAP-0429	Actions taken due to the audit storage failure.	The identity of the Security Administrator performing the function.
FAU_STG.1-NIAP-0429	None.	
FAU_STG.3	Actions taken due to exceeding the audit threshold.	The identity of the Security Administrator performing the function.
FCS_BCM_(EXT).1	None.	

<b>Requirement</b>	<b>Auditable Events</b>	<b>Audit Record Contents</b>
FCS_CKM.1(1)	a) Failure of the activity; b) Generation and loading of key.	Identify the failed activity and the data that caused the failure.
FCS_CKM.1(2)	a) Failure of the activity; b) Generation and loading of key pair for digital signatures.	Identify the failed activity and the data that caused the failure.
FCS_CKM.2	a) Failure of the activity; b) Generation and loading of key.	Identify the failed activity and the data that caused the failure.
FCS_CKM.4	a) Failure of the activity; b) Generation and loading of key.	Identify the failed activity and the data that caused the failure.
FCS_CKM_(EXT).2	a) Failure of the activity; b) Generation and loading of key.	Identify the failed activity and the data that caused the failure.
FCS_COP.1(1)	Failure of cryptographic operation.	Type of cryptographic operation.  Any applicable cryptographic mode(s) of operation, excluding any sensitive information.

<b>Requirement</b>	<b>Auditable Events</b>	<b>Audit Record Contents</b>
FCS_COP.1(2)	Failure of cryptographic operation.	Type of cryptographic operation Any applicable cryptographic mode(s) of operation, excluding any sensitive information.
FCS_COP.1(3)	Failure of cryptographic operation.	Type of cryptographic operation. Any applicable cryptographic mode(s) of operation, excluding any sensitive information.
FCS_COP.1(4)	Failure of cryptographic operation.	Type of cryptographic operation. Any applicable cryptographic mode(s) of operation, excluding any sensitive information.
FCS_COP_(EXT).1	Failure of cryptographic operation.	Type of cryptographic operation. Any applicable cryptographic mode(s) of operation, excluding any sensitive information.

<b>Requirement</b>	<b>Auditable Events</b>	<b>Audit Record Contents</b>
FCS_IKE_(EXT).1	a) Generation and loading of key pair for digital signatures; b) Changes to the pre-shared key used for authentication; c) All modifications to the key lifetimes; d) Failure of the authentication in Phase 1; e) Failure to negotiate a security association in Phase 2.	If failure occurs, record a descriptive reason for the failure.
FDP_IFC.1(1)	None.	
FDP_IFC.1(2)	None.	
FDP_IFF.1(1)	a) Decisions to permit or deny information flows; b) Operation applied to each information flow permitted.	Presumed identity of source subject. Identity of destination subject. Transport layer protocol, if applicable. Source subject service identifier, if applicable. Destination subject service identifier, if applicable. Identity of the interface on which the TOE received the packet. For denied information flows, the reason for denial.

<b>Requirement</b>	<b>Auditable Events</b>	<b>Audit Record Contents</b>
FDP_IFF.1(2)	Decisions to permit or deny information flows.	<p>Presumed identity of source subject.</p> <p>Identity of destination subject.</p> <p>Transport layer protocol, if applicable.</p> <p>Source subject service identifier, if applicable.</p> <p>Destination subject service identifier, if applicable.</p> <p>Identity of the interface on which the TOE received the packet.</p> <p>For denied information flows, the reason for denial.</p>
FDP_RIP.2	None	
FIA_AFL.1	a) The reaching of the threshold for the unsuccessful authentication attempts and the actions (e.g., disabling of an account) taken and the subsequent, if appropriate, restoration to the normal state (e.g., re-enabling of a terminal).	<p>Identity of the unsuccessful authentication attempts,</p> <p>Terminal identification,</p> <p>Action taken.</p>
FIA_ATD.1(1)	None.	
FIA_ATD.1(2)	None.	

<b>Requirement</b>	<b>Auditable Events</b>	<b>Audit Record Contents</b>
FIA_UAU.2	a) Successful and unsuccessful use of authentication mechanisms; b) All use of the authentication mechanism.	Claimed identity of the user using the authentication mechanism.  Success or failure of the authentication mechanism.
FIA_UAU_(EXT).5	a) The final decision on authentication; b) The result of each activated mechanism together with the final decision.	Claimed identity of the user attempting to authenticate.
FIA_UID.2	a) Unsuccessful use of the user identification mechanism, including the user identity provided; b) All use of the user identification mechanism, including the user identity provided (that is, those that authenticate to the TOE).	Claimed identity of the user using the identification mechanism.
FIA_USB.1	a) Unsuccessful binding of user security attributes to a subject (e.g., creation of a subject). b) Success and failure of binding of user security attributes to a subject.	The identity of the user whose attributes are attempting to be bound.
FMT_MOF.1(1)	All modifications in the behavior of the functions in the TSF.	The identity of the administrator performing the function, the function being performed and the data being used to perform the function (if available).

<b>Requirement</b>	<b>Auditable Events</b>	<b>Audit Record Contents</b>
FMT_MOF.1(2)	a) Enabling or disabling of the key-generation self-tests. b) All modifications in the behavior of the functions in the TSF.	The identity of the administrator performing the function, the function being performed and the data being used to perform the function (if available).
FMT_MOF.1(3)	All modifications in the behavior of the functions in the TSF.	The identity of the administrator performing the function, the function being performed and the data being used to perform the function (if available).
FMT_MOF.1(4)	All modifications in the behavior of the functions in the TSF.	The identity of the administrator performing the function, the function being performed and the data being used to perform the function (if available).
FMT_MOF.1(5)	All modifications in the behavior of the functions in the TSF.	The identity of the administrator performing the function, the function being performed and the data being used to perform the function (if available).
FMT_MOF.1(6)	All modifications in the behavior of the functions in the TSF.	The identity of the administrator performing the function, the function being performed and the data being used to perform the function (if available).
FMT_MOF.1(7)	All modifications in the behavior of the functions in the TSF.	The identity of the administrator performing the function, the function being performed and the data being used to perform the function (if available).

<b>Requirement</b>	<b>Auditable Events</b>	<b>Audit Record Contents</b>
FMT_MSA.1(1)	All manipulation of the security attributes.	The identity of the administrator performing the function, the function being performed and the security attributes being used to perform the function (if available).
FMT_MSA.1(2)	All manipulation of the security attributes.	The identity of the administrator performing the function, the function being performed and the security attributes being used to perform the function (if available).
FMT_MSA.3(1)	a) Modifications of the default setting of permissive or restrictive rules; b) All modifications of the initial values of security attributes.	The identity of the administrator performing the function, the function being performed and the security attributes being used to perform the function (if available).
FMT_MSA.3(2)	a) Modifications of the default setting of permissive or restrictive rules; b) All modifications of the initial values of security attributes.	The identity of the administrator performing the function, the function being performed and the security attributes being used to perform the function (if available).
FMT_MTD.1(1)	All modifications of the values of TSF data by the administrator.	The identity of the administrator performing the function, the function being performed and the values of TSF data being modified during the performance the function (if available).

<b>Requirement</b>	<b>Auditable Events</b>	<b>Audit Record Contents</b>
FMT_MTD.1(2)	All modifications of the values of cryptographic security data by the cryptographic administrator.	The identity of the administrator performing the function, the function being performed and the values of TSF data being modified during the performance the function (if available).
FMT_MTD.1(3)	All modifications to the time and date used to form the time stamps by the administrator.	The identity of the administrator performing the function, the function being performed the values of data being modified and the modifying data used during the performance the function.
FMT_MTD.1(4)	All modifications to the information flow policy ruleset by the Security Administrator.	The identity of the administrator performing the function, the function being performed the values of data being modified and the modifying data used during the performance the function.
FMT_MTD.2(1)	a) All modifications of the limits on TSF data b) All modifications in the actions to be taken in case of violation of the limits.	The identity of the administrator performing the function, the function being performed the values of data being modified and the modifying data used during the performance the function.

<b>Requirement</b>	<b>Auditable Events</b>	<b>Audit Record Contents</b>
FMT_MTD.2(2)	a) All modifications of the limits on TSF data. b) All modifications in the actions to be taken in case of violation of the limits.	The identity of the administrator performing the function, the function being performed the values of data being modified and the modifying data used during the performance the function.
FMT_MTD.2(3)	a) All modifications of the limits on TSF data. b) All modifications in the actions to be taken in case of violation of the limits.	The identity of the administrator performing the function, the function being performed the values of data being modified and the modifying data used during the performance the function.
FMT_REV.1	a) Unsuccessful revocation of security attributes; b) All attempts to revoke security attributes.	List of security attributes that were attempted to be revoked. The identity of the administrator performing the function.
FMT_SMF.1	Use of the management functions.	The identity of the administrator performing the function.  Identify the management function being performed
FMT_SMR.2	a) Modifications to the group of users that are part of a role; b) Unsuccessful attempts to use a role due to given conditions on the roles.	User IDs which are associated with the modifications.  The identity of the administrator performing the function.

Requirement	Auditable Events	Audit Record Contents
FPT_FLS.1	Failure of the TSF.	Indication that the TSF has failed with the type of failure that occurred.
FPT_ITA.1	The absence of TSF data when required by a TOE.	Include the type of TSF data that was not available and the condition that was to ensure availability.
FPT_ITC.1	None.	
FPT_ITL.1	a) The detection of modification of transmitted TSF data. b) The action taken upon detection of modification of transmitted TSF data.	Identify the data that detected as modified and the action taken upon detection of the modification.
FPT_PRO_(EXT).1	None.	
FPT_RCV.2	a) The fact that a failure or service discontinuity occurred; b) Resumption of the regular operation; c) Type of failure or service discontinuity.	Identify the failure, and that the TSF was able to recover to a secure state. If it is not possible to recover, enter maintenance mode.
FPT_RPL.1 (including replay of authentication data notification from the authentication server)	Detected replay attacks.	Identity of the user that was the subject of the reply attack
FPT_STM.1	Changes to the time.	Identify that the time has been changed and the administrator that took the action.

<b>Requirement</b>	<b>Auditable Events</b>	<b>Audit Record Contents</b>
FPT_TST_(EXT).1	Execution of this set of TSF self tests and the results of the tests.	The identity of the administrator performing the test, if initiated by an administrator. Report any results from the test.
FPT_TST.1(1)	Execution of this set of TSF self tests for Cryptography and the results of the tests.	The identity of the administrator performing the test, if initiated by an administrator. Report any results from the test.
FPT_TST.1(2)	Execution of this set of TSF self tests for key generation and the results of the tests.	The identity of the administrator performing the test, if initiated by an administrator. Report any results from the test.
FRU_RSA.1(1)	<p>a) Rejection of allocation operation due to resource limits.</p> <p>b) All attempted uses of the resource allocation functions for resources that are under control of the TSF.</p>	Identify the controlled resources (transport-layer quotas) that caused the rejection, and the source subject identifier.
FRU_RSA.1(2)	<p>a) Rejection of allocation operation due to resource limits.</p> <p>b) All attempted uses of the resource allocation functions for resources that are under control of the TSF.</p>	Identify the controlled resources (controlled connection-oriented resources) that caused the rejection, and the user.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	The identity of the user associated with the session that was terminated.
FTA_TAB.1	None.	

<b>Requirement</b>	<b>Auditable Events</b>	<b>Audit Record Contents</b>
FTA_TSE.1	<p>a) Denial of a session establishment due to the session establishment mechanism.</p> <p>b) All attempts at establishment of a user session.</p>	<p>The identity of the user attempting to establish the session.</p> <p>For unsuccessful attempts, the reason for denial of the establishment attempt.</p>
FTP_ITC.1(1)	<p>a) Failure of the trusted channel functions.</p> <p>b) Identification of the initiator and target of failed trusted channel functions.</p> <p>c) All attempted uses of the trusted channel functions.</p> <p>d) Identifier of the initiator and target of all trusted channel functions.</p>	<p>Indicated that the trusted channel failed and identification of the initiator and target of all trusted channels.</p>
FTP_ITC.1(2)	<p>a) Failure of the trusted channel functions.</p> <p>b) Identification of the initiator and target of failed trusted channel functions.</p> <p>c) All attempted uses of the trusted channel functions.</p> <p>d) Identifier of the initiator and target of all trusted channel functions.</p>	<p>Indicated that the trusted channel failed and identification of the initiator and target of all trusted channels.</p>

Requirement	Auditable Events	Audit Record Contents
FTP_TRP.1(1)	a) Failures of the trusted path functions. b) Identification of the user associated with all trusted path failures, if available. c) All attempted uses of the trusted path functions. d) Identification of the user associated with all trusted path invocations, if available.	Indicated that the trusted path failed and Identification of the claimed user identity.
FTP_TRP.1(2)	a) Failures of the trusted path functions. b) Identification of the user associated with all trusted path failures, if available. c) All attempted uses of the trusted path functions. d) Identification of the user associated with all trusted path invocations, if available.	Indicated that the trusted channel failed and Identification of the claimed user identity.

#### 5.1.1.4 User identity association (FAU\_GEN.2-NIAP-0410)

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.

#### 5.1.1.5 Potential violation analysis (FAU\_SAA.1-NIAP-0407)

FAU\_SAA.1.1-NIAP-0407 – The TSF shall be able to apply a set of rules in monitoring the audited events and based upon these rules indicate a potential violation of the TSP.

FAU\_SAA.1.2-NIAP-0407 The TSF shall enforce the following rules for monitoring audited events:

a) Accumulation or combination of [

1. Security Administrator specified number of user authentication failures;
2. Any detected replay of TSF data or security attributes;
3. Any failure of the cryptographic self-tests;
4. Any failure of the other TSF self-tests;
5. Cryptographic Administrator specified number of encryption failures;
6. Cryptographic Administrator specified number of decryption failures; and

] known to indicate a potential security violation;

b) [selection: [assignment: *any other rules*], "no additional rules"]

*Application Note: The intent of this requirement is that an alarm is generated (FAU\_ARP.1) once the threshold for an event is met. Once the alarm has been generated it is assumed that FAU\_ARP.1 resets the "count" for that event to zero. The Security Administrator settable number of authentication failures in (1) is intended to be the same value as specified in FIA\_AFL.1.1. Note that the user authentication failure is distinct from failure to authenticate data packets, e.g. via ESP or AH.*

*The failure of TSF self-tests in (4).*

#### 5.1.1.6 Audit review (FAU\_SAR.1)

FAU\_SAR.1.1 – The TSF shall provide [the Audit Administrators] with the capability to read [all audit data] from the audit records.

FAU\_SAR.1.2 – **Refinement:** The TSF shall provide the audit records in a manner suitable for the ~~user~~ Audit **Administrators** to interpret the information.

#### 5.1.1.7 Restricted audit review (FAU\_SAR.2)

FAU\_SAR.2.1 – **Refinement:** The TSF shall prohibit all users read access to the audit records in the audit trail, except **the Audit Administrators** ~~those users that have been granted explicit read access.~~

#### 5.1.1.8 Selectable audit review (FAU\_SAR.3)

FAU\_SAR.3.1 - The TSF shall provide the ability to perform *searches and sorting* of audit data based on:

a) [user identity;

- b) source subject identity;
  - c) destination subject identity;
  - d) ranges of one or more: dates, times, user identities, subject service identifiers, or transport layer protocol;
  - e) rule identity;
  - f) TOE network interfaces; and
  - g) [selection: [assignment: other criteria], no additional criteria]].
- 74 Application Note: Audit data should be capable of being searched and sorted on all criteria specified in a – g, if applicable (i.e., not all criteria will exist in all audit records). Sorting means to arrange the audit records such that they are “grouped” together for administrative review. For example, the Audit Administrator may want all the audit records for a specified source subject identity or range of source subject identities (e.g., IP source address or range of IP source addresses) presented together to facilitate their audit review. If no additional criteria are provided by the TOE to perform searches or sorting of audit data, the ST author selects “no additional criteria”.

#### 5.1.1.9 Audit event selection (FAU\_SEL.1-NIAP-0407)

FAU\_SEL.1.1-NIAP-0407 - **Refinement:** The TSF shall **allow only the Security 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]].
- 75 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.

#### 5.1.1.10 Protected audit trail storage (FAU\_STG.1-NIAP-0429)

FAU\_STG.1.1-NIAP-0429 – **Refinement:** The TSF shall **restrict the deletion of stored** audit records in the audit trail **to the Audit Administrator**.

FAU\_STG.1.2-NIAP-0429 – The TSF shall be able to *prevent* modifications to the audit records in the audit trail.

#### 5.1.1.11 Action in case of possible audit data loss (FAU\_STG.3)

FAU\_STG.3.1 - **Refinement:** The TSF shall [immediately alert the **administrators** by displaying a message at the local console, [and at the remote administrative console when an administrative session exists for each of the defined administrative roles, at the option of the Security Administrator generate an audible alarm] [assignment: other actions determined by the ST author] if the audit trail exceeds [a Security Administrator settable percentage of storage capacity].

76 Application Note: As with FAU\_ARP.1, the TSF provides a message to the local console regardless of whether an administrator is logged in. The message is displayed at the remote console if an administrator is already logged in, or when an administrator logs in. This requirement specifies that the message is sent to the first established session for each of the defined roles to ensure someone in the administrator staff is aware of the alert as soon as possible.

#### 5.1.1.12 Site-configurable Prevention of audit data loss (FAU\_STG.NIAP-0414-1-NIAP-0429)

FAU\_STG.NIAP-0414-1.1-NIAP-0429. **Refinement:** The TSF shall provide the **Security Administrator** the capability to select one or more of the following actions [selection: '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] to be taken if the audit trail is full.

FAU\_STG.NIAP-0414-1.2-NIAP-0429. **Refinement:** The TSF shall **enforce the Security Administrator's** [selection: choose one of: "ignore auditable events", "prevent auditable events, except those taken by the authorised 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.

77 Application Note: The TOE provides the security administrator the option of preventing audit data loss by preventing auditable events from occurring. The security administrator's actions under these circumstances are not required to be audited. The TOE also provides the security 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.

## 5.1.2 Cryptographic Support (FCS)

This section specifies the cryptographic support required in the TOE. Evolving public standards on cryptographic functions and related areas have required an interim approach to writing cryptographic requirements. These cryptographic requirements are expected to be achievable in commercial products in the near term, and gradually mature over time. Today these requirements represent a step in the direction of helping to improve the security in COTS products. Over time, the Protection Profile will be updated as the underlying public standards and the body of related special publications mature.

### 5.1.2.1 Extended: Baseline Cryptographic Module (FCS\_BCM\_(EXT))

The cryptographic requirements are structured to accommodate use of the FIPS 140-2 standard and NIST’s Cryptomodule Validation Program (CMVP) in meeting the requirements. Note that *FIPS-approved* cryptographic functions are required to be implemented in a *FIPS-validated module running in FIPS-approved mode*. FCS\_BCM reflects this requirement, and it specifies the required FIPS validation levels for the security functions. Note also that some of the requirements of this Protection Profile go beyond what is required for FIPS 140-2 validation.

*Application Note: A FIPS-approved cryptographic function is a security function (e.g., cryptographic algorithm, cryptographic key management technique, or authentication technique) that is either: 1) specified in a Federal Information Processing Standard (FIPS), or 2) adopted in a FIPS and specified either in an appendix to the FIPS or in a document referenced by the FIPS.*

#### Extended: Baseline Cryptographic Module (FCS\_BCM\_(EXT).1)

**FCS\_BCM\_(EXT).1.1** All FIPS-approved cryptographic functions implemented by the TOE shall be implemented in a cryptomodule that is FIPS 140-2 validated, and perform the specified cryptographic functions in a FIPS-approved mode of operation. The FIPS 140-2 validation shall include an algorithm validation certificate for all FIPS-approved cryptographic functions implemented by the TOE.

*Application Note: This Protection Profile shall use the term “FIPS 140-2” for simplicity. FIPS PUB 140-2 is currently undergoing a regular five year review; in the near future, FIPS PUB 140-3 will supersede it. Security Targets written to comply with this Protection Profile may replace it with the successor standard that is in force at the time of evaluation.*

*Application Note: This requirement does not preclude additional cryptographic algorithms from being implemented in the cryptomodule, and/or used by the TOE for purposes OTHER than those explicitly stated in this Protection Profile.*

**FCS\_BCM\_(EXT).1.2** All cryptographic modules implemented in the TOE  
**[selection:**

- (1) Entirely in hardware shall have a minimum overall rating of FIPS PUB 140-2, Level 3,**
- (2) Entirely in software shall have a minimum overall rating of FIPS PUB 140-2, Level 1 and also meet FIPS PUB 140-2, Level 3 for the following: Cryptographic Module Ports and Interfaces; Roles, Services and Authentication; Cryptographic Key Management; and Design Assurance.**
- (3) As a combination of hardware and software shall have a minimum overall rating of FIPS PUB 140-2, Level 1 and also meet FIPS PUB 140-2, Level 3 for the following: Cryptographic Module Ports and Interfaces; Roles, Services and Authentication; Cryptographic Key Management; and Design Assurance. ]**

*Application Note: “Combination of hardware and software” means that some part of the cryptographic functionality will be implemented as a software component of the TSF. The combination of a cryptographic hardware module and a software device driver whose sole purpose is to communicate with the hardware module is considered a hardware module rather than “combination of hardware and software”.*

*Application Note: Note that the requirements for selections (2) and (3) are the same. The ST author should make it clear how the cryptomodule is implemented.*

### **5.1.2.2 Cryptographic Key Management (FCS\_CKM)**

NIST Special Publication 800-57, “Recommendation for Key Management” contains additional protection mechanisms that vendors are encouraged to implement. It should also be used as guidance for the cryptographic key management requirements.

#### **Cryptographic Key Generation (for symmetric keys) (FCS\_CKM.1(1))**

**FCS\_CKM.1.1(1) Refinement:** The TSF shall generate symmetric cryptographic keys using a **FIPS-Approved Random Number Generator as specified in FCS\_COP\_(EXT).1, and provide integrity protection to generated symmetric keys in accordance with NIST SP 800-57 “Recommendation for Key Management” Section 6.1.**

*Application Note: NIST SP 800-57 “Recommendation for Key Management” Section 6.1 states: “Integrity protection can be provided by cryptographic integrity mechanisms (e.g. cryptographic checksums, cryptographic hashes, MACs, and signatures), non-cryptographic integrity mechanisms (e.g. CRCs, parity, etc.) [...], or physical protection mechanisms.” Guidance for the selection of appropriate integrity mechanisms is given in Sections 6.2.1.2 and 6.2.2.2 of NIST SP 800-57 “Recommendation for Key Management”.*

*Application Note: Note that there is a separate requirement for Cryptographic Key Agreement (FCS\_COP.1(4)).*

#### **Cryptographic Key Generation (for asymmetric keys) (FCS\_CKM.1(2))**

**FCS\_CKM.1.1(2) Refinement:** The TSF shall generate **asymmetric** cryptographic keys in accordance with the **mathematical specifications of the FIPS-approved or NIST-recommended standard [assignment: specify standard(s)]**, using a domain parameter generator and **[selection:**

**(1) a FIPS-Approved Random Number Generator as specified in FCS\_COP\_(EXT).1, and/or**

**(2) a prime number generator as specified in ANSI X9.80 “Prime Number Generation, Primality Testing, and Primality Certificates” using random integers with deterministic tests, or constructive generation methods ]**

in a cryptographic key generation scheme that meets the following:

- The TSF shall provide integrity protection and assurance of domain parameter and public key validity to generated asymmetric keys in accordance with NIST SP 800-57 “Recommendation for Key Management” Section 6.1.
- Generated key strength shall be equivalent to, or greater than, a symmetric key strength of 128 bits using conservative estimates.

*Application Note: NIST SP 800-57 “Recommendation for Key Management” Section 6.1 states: “Integrity protection can be provided by cryptographic integrity mechanisms (e.g. cryptographic checksums, cryptographic hashes, MACs, and signatures), non-cryptographic integrity mechanisms (e.g. CRCs, parity, etc.) [...], or physical protection mechanisms.” Guidance for the selection of appropriate integrity mechanisms is given in Sections 6.2.1.2 and 6.2.2.2 of NIST SP 800-57 “Recommendation for Key Management”.*

*Application Note: Assurance of domain parameter and public key validity provides confidence that the parameters and keys are arithmetically correct. Guidance for the selection of appropriate validation mechanisms is given in NIST SP 800-57 “Recommendation for Key Management,” NIST Special Publication 800-56A, “Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography,” and FIPS PUB 186-3, “Digital Signature Standard.”*

*Application Note: See NIST Special Publication 800-57, “Recommendation for Key Management” for information about equivalent key strengths.*

## **Cryptographic Key Distribution (FCS\_CKM.2)**

**FCS\_CKM.2.1** The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method **[selection:**

**(3) Manual (Physical) Method, and/or**

**(4) Automated (Electronic) Method ]**

that meets the following:

- **NIST Special Publication 800-57, “Recommendation for Key Management” Section 8.1.5**

- **NIST Special Publication 800-56A, “Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography”**

*Application Note: NIST Special Publication 800-56A “Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography” is only applicable when public key schemes are used in key transport methods.*

*Application Note: DoD applications may have additional key distribution requirements related to the DoD PKI and certificate formats.*

**Extended: Cryptographic Key Handling and Storage (FCS\_CKM\_(EXT).2)**

**FCS\_CKM\_(EXT).2.1** The TSF shall perform a key error detection check on each transfer of key (internal, intermediate transfers).

*Application Note: A parity check is an example of a key error detection check.*

**FCS\_CKM\_(EXT).2.2** The TSF shall store persistent secret and private keys when not in use in encrypted form or using split knowledge procedures.

*Application Note: Note that this requirement is stronger than the FIPS 140-2 key storage requirements, which state: “Cryptographic keys stored within a cryptographic module shall be stored in plaintext form or encrypted form.”*

*Application Note: A persistent key, such as a file encryption key, is one that must be available in the system over long periods of time. A non-persistent key, such as a key used to encrypt or decrypt a single message or a session, is one that is ephemeral in the system.*

*Application Note: “When not in use” is interpreted in the strictest sense so that persistent keys only exist in plaintext form during intervals of operational necessity. For example, a file encryption key exists in plaintext form only during actual encryption and/or decryption processing of a file. Once the file is decrypted or encrypted, the file encryption key should immediately be covered for protection.*

*Application Note: A “split knowledge procedure” is a process by which a cryptographic key is split into multiple key components, individually sharing no knowledge of the original key, which can be subsequently input into, or output from, a cryptographic module by separate entities and combined to recreate the original cryptographic key.*

**FCS\_CKM\_(EXT).2.3** The TSF shall destroy non-persistent cryptographic keys after a cryptographic administrator-defined period of time of inactivity.

*Application Note: The cryptographic administrator must have the ability to set a threshold of inactivity after which non-persistent keys must be destroyed in accordance with FCS\_CKM.4.*

**FCS\_CKM\_(EXT).2.4** The TSF shall prevent archiving of expired (private) signature keys.

*Application Note: This requirement is orthogonal to typical system back-up procedures. Therefore, it does not address the problem of archiving an active (private) signature key during a system back-up and saving the key beyond its intended life span.*

### **Cryptographic Key Destruction (FCS\_CKM.4)**

*Application Note: Note that this requirement is stronger than the FIPS 140-2 key zeroization requirements, which state: “A cryptographic module shall provide methods to zeroize all plaintext secret and private cryptographic keys and CSPs within the module.”*

**FCS\_CKM.4.1 Refinement:** The TSF shall destroy cryptographic keys in accordance with a **cryptographic key zeroization method** that meets the following:

- a) **Key zeroization requirements of FIPS PUB 140-2, “Security Requirements for Cryptographic Modules”**
- b) **Zeroization of all plaintext cryptographic keys and all other critical cryptographic security parameters shall be immediate and complete.**

*Application Note: The term “immediate” here is meant to impart some urgency to the destruction: it should happen as soon as practical after the key is no longer required to be in plaintext. It is certainly permissible to complete a critical section of code before destroying the key. However, the destruction shouldn’t wait for idle time, and there shouldn’t be any non-determined event (such as waiting for user input) which occurs before it is destroyed.*

- c) **The TSF shall zeroize each intermediate storage area for plaintext key/critical cryptographic security parameter (i.e., any storage, such as memory buffers, that is included in the path of such data) upon the transfer of the key/critical cryptographic security parameter to another location.**

*Application Note: Item c) pertains to the elimination of internal, temporary copies of keys/parameters during processing, and not to the locations that are used for the storage of the keys, which are specified in item b). The temporary locations could include memory registers, physical memory locations, and even page files and memory dumps.*

- d) **For non-volatile memories other than EEPROM and Flash, the zeroization shall be executed by overwriting three or more times using a different alternating data pattern each time.**

*Application Note: Although verification of the zeroization of each intermediate location consisting of non-volatile memories is desired here (by checking for the final known alternating data pattern), it is not required at this time. However, vendors are highly encouraged to incorporate this verification whenever possible into their implementations.*

- e) **For volatile memory and non-volatile EEPROM and Flash memories, the zeroization shall be executed by a single direct overwrite consisting of a pseudo random pattern, followed by a read-verify.**

### **5.1.2.3 Cryptographic Operation (FCS\_COP)**

**Cryptographic Operation (for data encryption/decryption)  
(FCS\_COP.1(1))**

**FCS\_COP.1.1(1) Refinement:** The cryptomodule shall perform **encryption and decryption using the FIPS-approved security function AES algorithm operating in [assignment: one or more FIPS-approved modes] and cryptographic key size of [selection: one or more of 128 bits, 192 bits, 256 bits].**

**Cryptographic Operation (for cryptographic signature) (FCS\_COP.1(2))**

**FCS\_COP.1.1(2) Refinement:** The TSF shall perform **cryptographic signature services using the FIPS-approved security function [selection:**

**(5) Digital Signature Algorithm (DSA) with a key size (modulus) of [assignment: 2048 bits or greater],**

**(6) RSA Digital Signature Algorithm (rDSA) with a key size (modulus) of [assignment: 2048 bits or greater], or**

**(7) Elliptic Curve Digital Signature Algorithm (ECDSA) with a key size of [selection: one or more of 256 bits, 384 bits, 521 bits], using only the NIST curve(s) [selection: one or more of P-256, P-384, P-521 as defined in FIPS PUB 186-3, "Digital Signature Standard"] ]**

**that meets NIST Special Publication 800-57, "Recommendation for Key Management."**

*Application Note: For elliptic curve-based schemes, the key size refers to the  $\log_2$  of the order of the base point. As the preferred approach for key exchange, elliptic curves will be required after all the necessary standards and other supporting information are fully established.*

**Cryptographic Operation (for cryptographic hashing) (FCS\_COP.1(3))**

**FCS\_COP.1.1(3) Refinement:** The TSF shall perform **cryptographic hashing services using the FIPS-approved security function Secure Hash Algorithm and message digest size of [selection: one or more of 256 bits, 384 bits, 512 bits].**

*Application Note: The message digest size should correspond to double the system symmetric encryption key strength.*

**Cryptographic Operation (for cryptographic key agreement) (FCS\_COP.1(4))**

*Application Note: "Cryptographic key agreement" is a procedure where the resultant secret keying material is a function of information contributed by two participants, so that no party can predetermine the value of the secret keying material independently from the contributions of the other parties.*

**FCS\_COP.1.1(4) Refinement:** The TSF shall perform **cryptographic key agreement services using the FIPS-approved security function as specified in NIST Special Publication 800-56A, "Recommendation for Pair-Wise Key Establishment Schemes**

**Using Discrete Logarithm Cryptography” [selection:**

- (1) [assignment: Finite Field-based key agreement algorithm] and cryptographic key sizes (modulus) of [assignment: 2048 bits or greater], or**
- (2) [assignment: Elliptic Curve-based key agreement algorithm] and cryptographic key size of [assignment: one or more of 256 bits, 384 bits, 521 bits], using only the NIST curve(s) [selection: one or more of P-256, P-384, P-521 as defined in FIPS PUB 186-3, “Digital Signature Standard”]**

*Application Note: For elliptic curve-based schemes, the key size refers to the  $\log_2$  of the order of the base point. As the preferred approach for key exchange, elliptic curves will be required after all the necessary standards and other supporting information are fully established.*

**that meets NIST Special Publication 800-57, “Recommendation for Key Management.”**

*Application Note: Some authentication mechanism on the keying material is recommended. In addition, repeated generation of the same shared secrets should be avoided.*

*Application Note: FIPS 140-2 Annex D specifies references for FIPS-approved Key Establishment Techniques, one of which is NIST Special Publication 800-56A, “Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography.”*

**Extended: Random Number Generation (FCS\_COP\_(EXT).1)**

**FCS\_COP\_(EXT).1.1 The TSF shall perform all random number generation (RNG) services in accordance with a FIPS-approved RNG [assignment: one of the RNGS specified in FIPS 140-2 Annex C] seeded by [selection:**

- (1) one or more independent hardware-based entropy sources, and/or**
- (2) one or more independent software-based entropy sources, and/or**
- (3) a combination of hardware-based and software-based entropy sources. ]**

*Application Note: The ST author should specify how the RNG is seeded.*

**FCS\_COP\_(EXT).1.2 The TSF shall defend against tampering of the random number generation (RNG)/ pseudorandom number generation (PRNG) sources.**

*Application Note: The RNG/PRNG should be resistant to manipulation or analysis of its sources, or any attempts to predictably influence its states. Three examples of very different approaches the TSF might pursue to address this include: a) identifying the fact that physical security must be applied to the product, b) applying checksums over the sources, or c) designing and implementing the TSF RNG with a concept similar to a keyed hash (e.g., where periodically, the initial state of the hash is*

*changed unpredictably and each change is protected as when provided on a tamper-protected token, or in a secure area of memory.*

#### 5.1.2.4 Internet key exchange (FCS\_IKE\_(EXT).1)

FCS\_IKE\_(EXT).1.1 The TSF shall provide cryptographic key establishment techniques in accordance with RFC 2409 as follows(s):

Phase 1, the establishment of a secure authenticated channel between the TOE and another remote router endpoint, shall be performed using one of the following, as configured by the security administrator:

- Main Mode
- Aggressive Mode]

New Group mode shall include the private group 14, 2048-bit MOD P, [*selection:[assignment: other group modes determined by the ST author, ]'no other group modes''*] for the Diffie-Hellman key exchange.

Phase 2, negotiation of security services for IPsec, shall be done using Quick Mode, using SHA-1 as the pseudo-random function. Quick Mode shall generate key material that provides perfect forward secrecy.

FCS\_IKE\_(EXT).1.2 The TSF shall require the **nonce, and the x of  $g^{xy}$**  be randomly generated using FIPS-approved random number generator when computation is being performed.

- The recommended nonce sizes are to be between 8 and 256 bytes;
- The minimum size for the x should be 256 bits.

FCS\_IKE\_(EXT).1.3 When performing authentication using pre-shared keys, the key shall be generated using the FIPS approved random number generator specified in FCS\_COP\_(EXT).1 .1.

FCS\_IKE\_(EXT).1.4 The TSF shall compute the value of SKEYID (as defined in RFC 2409), using a NIST-approved hashing function as the pseudo-random function. The TSF shall be capable of authentication using the methods for

- Signatures:  $SKEYID = sha(Ni\_b \mid Nr\_b, g^{xy})$
- Pre-shared keys:  $SKEYID = sha(\text{pre-shared-key}, Ni\_b \mid Nr\_b)$

- [*selection: Authentication using Public key encryption, computing SKEYID as follows:*  $SKEYID = sha(sha(Ni\_b \parallel Nr\_b), CKY-I \parallel CKY-R)$ , [*assignment: other authentication method, "no other authentication methods"*]]

78 Application Note: If public key encryption is the method of choice, the sha algorithm listed in the requirement will be used. If another option is selected, a different authentication method or a different hash algorithm for generating SKEYID may be specified.

79 Refer to RFC 2409 for an explanation of the notation and definitions of the terms.

FCS\_IKE\_(EXT).1.5 The TSF shall compute authenticated keying material as follows:

- $SKEYID\_d = sha(SKEYID, g^{xy} \parallel CKY-I \parallel CKY-R \parallel 0)$
- $SKEYID\_a = sha(SKEYID, SKEYID\_d \parallel g^{xy} \parallel CKY-I \parallel CKY-R \parallel 1)$
- $SKEYID\_e = sha(SKEYID, SKEYID\_a \parallel g^{xy} \parallel CKY-I \parallel CKY-R \parallel 2)$
- [*selection: [assignment: other methods for computing the authenticated keying material], none*]]

80 Application Note: If the assignment is selected, a different method for computing the authenticated keying material may be used, or a different hash algorithm may be specified.

FCS\_IKE\_(EXT).1.6 To authenticate the Phase 1 exchange, the TSF shall generate HASH\_I if it is the initiator, or HASH\_R if it is the responder as follows:

- $HASH\_I = sha(SKEYID, g^{xi} \parallel g^{xr} \parallel CKY-I \parallel CKY-R \parallel SAi\_b \parallel IDii\_b)$
- $HASH\_R = sha(SKEYID, g^{xr} \parallel g^{xi} \parallel CKY-R \parallel CKY-I \parallel SAi\_b \parallel IDir\_b)$

81 Application Note: Refer to RFC 2409 for an explanation of the notation and definitions of the terms.

FCS\_IKE\_(EXT).1.7 The TSF shall be capable of authenticating IKE Phase 1 using the following methods as defined in RFC 2409, as configured by the security administrator:

- Authentication with digital signatures:** The TSF shall use [*selection: RSA, DSA,* [*selection: [assignment: other digital signature algorithms], "no other digital signature algorithms"*]]
- when an RSA signature is applied to HASH I or HASH R it must be first PKCS#1 encoded. The TSF shall check the HASH\_I and HASH\_R values sent against a

computed value to detect any changes made to the proposed transform negotiated in the phase one. If changes are detected, the session shall be terminated and an alarm shall be generated.

- c) [selection:[assignment: X.509 certificates Version 3 [selection: other version of X.509 certificates, “no other versions”]] X.509 V3 implementations, if implemented, shall be capable of checking for validity of the certificate path, and at option of SA, check for certificate revocation using [selection: CRL, OCSP, SVCP].
- d) **Authentication with a pre-shared key:** The TSF shall allow authentication using a pre-shared key.

FCS\_IKE\_(EXT).1.8 The TSF shall compute the hash values for Quick Mode in the following way

$$\text{HASH}(1) = \text{sha}(\text{SKEYID\_a}, \text{M-ID} \mid (\text{assignment: any ISAKMP payload after HASH}(1) \text{ header contained in the message}))$$

$$\text{HASH}(2) = \text{sha}(\text{SKEYID\_a}, \text{M-ID} \mid \text{Ni\_b} \mid (\text{assignment: any ISAKMP payload after HASH}(2) \text{ header contained in the message}))$$

$$\text{HASH}(3) = \text{sha}(\text{SKEYID\_a}, 0 \mid \text{M-ID} \mid \text{Ni\_b} \mid \text{Nr\_b})$$

- 82 Application Note: The following steps will be performed when using the HASH computation:  
 initiator computes HASH(1) and sends to responder  
 responder validates computation of HASH(1) and computes HASH(2) and sends HASH(2) to initiator  
 initiator validates computation of HASH(2) and computes HASH(3) and sends HASH(3) to responder
- 83 IKE is only optional when Security Association (SA) elects not to use perfect forward secrecy.
- 84 Verifying that a TFS implementation actually checks HASH(1), HASH(2), and HASH(3) values sent against a computed value is important in detecting changes that could have been made to propose transform negotiated in Quick Mode (not as likely as Phase One because Quick Mode is encrypted).
- 85 The ordering of the ISAKMP payloads may differ because Quick Mode only specifies the location of the HASH and SA payload.

FCS\_IKE\_(EXT).1.9 The TSF shall compute new keying material during Quick Mode as follows:

[selection: when using perfect forward secrecy

KEYMAT = sha(SKEYID\_d, g(qm)<sup>xy</sup> | protocol | SPI | Ni\_b | Nr\_b),

When perfect forward secrecy is not used

KEYMAT = sha(SKEYID\_d | protocol | SPI | Ni\_b | Nr\_b)]

FCS\_IKE\_(EXT).1.10 The TSF shall at a minimum, support the following ID types:

[assignment: ID\_IPV4\_ADDR, ID\_FQDN, ID\_USER\_FQDN,  
ID\_IPV4\_ADDR\_SUBNET, ID\_IPV6\_ADDR, ID\_IPV6\_ADDR\_SUBNET,  
ID\_IPV4\_ADDR\_RANGE, ID\_IPV6\_ADDR\_RANGE, ID\_DER\_ASN1\_DN,  
ID\_DER\_ASN1\_GN, ID\_KEY\_ID].

- 86 Application Note: It should be noted that the Internet Protocol Version 6(IPv6) Interim Transition Guidance memorandum, September 29, 2003, provides support to begin to procure/acquire IPv6 capable GIG assets on October 1, 2003 and a goal for complete transition to IPv6 at FY2008

### 5.1.3 User Data Protection (FDP)

#### 5.1.3.1 Subset information flow control (unauthenticated policy) (FDP\_IFC.1(1) )

FDP\_IFC.1.1(1) - The TSF shall enforce the [UNAUTHENTICATED INFORMATION FLOW SFP] on [source subject: TOE interface on which information is received; destination subject: TOE interface to which information is destined; information: network packets; and operations: pass information by opening a relay connection through the TSF on behalf of the source subject to the destination subject, and with the TSF ensuring the following conditions:

- a) the connection from the source subject is from a valid peer network,
- b) the new relay connection is established to the destination subject on a valid peer network.]

#### 5.1.3.2 Subset information flow control (authenticated policy) (FDP\_IFC.1(2))

FDP\_IFC.1.1(2) The TSF shall enforce the [AUTHENTICATED INFORMATION FLOW SFP] on [source subject representing authenticated peer routers: source network identifier; destination subject: TOE interface to which information is destined; information: network packets; and operations: pass by opening a relay connection from the TSF on behalf of the source subject to the destination subject, and with the TSF ensuring the following conditions:

- a) the connection from the source subject is from a valid peer network,
- b) the new relay connection is established to the destination subject on a valid peer network.]

### 5.1.3.3 Simple Security attributes (unauthenticated policy) (FDP\_IFF.1(1))

FDP\_IFF.1.1(1) - The TSF shall enforce the [UNAUTHENTICATED INFORMATION FLOW SFP] based on the following types of subject and information security attributes:

- a) [Source subject security attributes:
  - set of source entity identifiers; and
  - [selection: [assignment: other subject security attributes], none].
- b) Destination subject security attributes:
  - Set of destination entity identifiers; and
  - [selection: [assignment: other subject security attributes], none].

87 Application Note: For the entities, the administrator knows the set of identifiers that can be associated with the physical router interfaces; therefore, they are not “presumed” identifiers. The term “identifiers” was used instead of “addresses” to allow for technologies that are not address-based (e.g., circuit identifiers instead of source and destination addresses).

88 The ST author should specify other attributes that are used to identify the source and destination entity sets, based on the technology implemented by the TOE.

- c) Information security attributes:
  - presumed identity of source entity<sup>1</sup>;
  - identity of destination entity;

---

<sup>1</sup> The TOE can make no claim as to the real identity of any source entity; the TOE can only suppose that such identities are accurate. Therefore, a “presumed identity” is used to identify source entities. Note, however, that the TOE can ensure that the identity is included in the set that is associated with the interface (see FDP\_IFF.1.6(1)).

- transport layer protocol;
- source entity service identifier;
- destination entity service identifier (e.g., TCP or User Datagram Protocol (UDP) destination port number);

89 Application Note: The transport layer protocol is what is specified in the 8-bit protocol field in the IP header (e.g., this would include ICMP and is not limited to TCP or UDP). The concept of a “service identifier” may differ depending on the networking stack used; the intent is to specify a service that is above the network and transport layers in the protocol stack. A “service” in the IP stack would be NTP, Trivial File Transfer Protocol (TFTP), etc.

90 Application Note: Not all of the above security attributes will exist in all network packets. However, the TOE’s ruleset allows the Security Administrator to select and filter on any of the above security attributes as part of the policy decision. The intent is that if a network packet includes any of the above security attributes, those attributes may be used in the policy decision. The ST author should fill in the assignment all attributes that the Security Administrator is able to specify when creating the router rules.

for non-IP-based network stacks: [assignment: information security attributes]].

91 Application Note: If a compliant TOE uses an IP based network stack (including IP running on top of another protocol, such as Asynchronous Transfer Mode (ATM)), then the first selection is made. If the TOE uses a network stack that is not IP-based (e.g., ATM without IP) then the ST author uses the second selection and fills in the assignment with the attributes that provide a commensurate level of confidence for the protocols employed that network packets can be correctly associated with a connection.

FDP\_IFF.1.2(1) - **Refinement:** The TSF shall permit an information flow between a **source** entity and a **destination entity** via a controlled operation if the following rules hold:

- [the presumed identity of the source entity is in the set of source entity identifiers;
- the identity of the destination entity is in the set of destination entity identifiers;
- the information security attributes match the attributes in an information flow policy rule (contained in the information flow policy ruleset defined by the Security Administrator) according to the following algorithm [assignment: algorithm used by the TOE to match information security attributes to information flow policy rules]; and

- the selected information flow policy rule specifies that the information flow is to be permitted].

92 Application Note: In a router, the administrator specifies information flow policy rules that contain information security attribute values (or wildcards that “stand” for multiple values of the same type; e.g., 127.\*.\* would represent any IP address that begins with “127”), and associate with that rule an action that permits the information flow or disallows the information flow. When a packet arrives at the source interface, the information security attribute values of the packet are compared to each information flow policy rule by some TOE-specified algorithm, and when a match is found the action specified by that rule is taken. Since wildcards would allow the specific attributes in a packet to potentially match more than one rule, the ST author needs to fill in the assignment with the algorithm the TOE uses to find a matching rule. This could be “first match”, “most specific match”, or some more elaborate description.

FDP\_IFF.1.3(1) - The TSF shall enforce the [assignment: additional information flow control SFP rules].

FDP\_IFF.1.4(1) - The TSF shall provide the following [the Security Administrator shall have the capability to view all information flows allowed by the information flow policy ruleset before the ruleset is applied].

93 Application Note: Some routers create additional rules as a side-effect of creating a rule; for example, a router may create a rule allowing a File Transfer Protocol (FTP) data channel when a rule allowing FTP (control connections) is created. This requirement allows an administrator to view the entire ruleset so that they can identify such rules and confirm that the ruleset reflects the desired policy.

94 “before the rule set is applied” means that the administrator is able to view the entire rule set before it is put into use on the TOE. This gives the administrator the opportunity to address any errors or unintended flows.

FDP\_IFF.1.5(1) - The TSF shall explicitly authorize an information flow based on the following rules: [none].

FDP\_IFF.1.6(1) - The TSF shall explicitly deny an information flow based on the following rules:

- [The TOE shall reject requests for access or services where the presumed source identity of the information received by the TOE is not included in the set of source identifiers for the source subject;

95 Application Note: The intent of this requirement is to ensure that a user cannot send packets originating on one TOE interface claiming to originate on another TOE interface.

- The TOE shall reject requests for access or services where the presumed source identity of the information received by the TOE specifies a broadcast identity;

96 Application Note: A broadcast identity is one that specifies more than one host address on a network. It is understood that the TOE can only know the sub-netting configuration of networks directly connected to the TOE's interfaces and therefore can only be aware of broadcast addresses on those networks.

- The TOE shall reject requests for access or services where the presumed source identity of the information received by the TOE specifies a loopback identifier;
- The TOE shall reject requests in which the information received by the TOE contains the route (set of host network identifiers) by which information shall flow from the source subject to the destination subject].

#### 5.1.3.4 Simple security attributes (authenticated policy) (FDP\_IFF.1(2))

FDP\_IFF.1.1(2) - The TSF shall enforce the [AUTHENTICATED INFORMATION FLOW SFP] based on the following types of subject and information security attributes:

a) [Source subject security attributes:

- source network identifier; and
- [selection: [assignment: other subject security attributes], none].

97 Application Note: Note that the above does not require a "userid" (distinct from the network identifier for the subject) to be specified for use in information flow policy rule; if the TOE provides this feature then it should be listed under the second bullet.

b) Destination subject security attributes:

- Set of destination network identifiers; and
- [selection: [assignment: other subject security attributes], none].

98 Application Note: The Security Administrator knows the set of identifiers that can be associated with the physical router interfaces; therefore, they are not "presumed" identifiers. The term "identifiers" was used instead of "addresses" to allow for technologies that are not address-based (e.g., circuit identifiers instead of source and destination addresses).

99 The ST author should specify other attributes that are used to identify the source subject and destination subject set, based on the technology implemented by the TOE.

c) Information security attributes:

- identity of source subject;
- identity of destination subject;
- transport layer protocol;
- destination subject service identifier (e.g., TCP destination port number);

100 Application Note: The concept of a “service identifier” may differ depending on the networking stack used; the intent is to specify a service that is above the network and transport layers in the protocol stack.

- [selection: [assignment: other information security attributes], none].

101 Application Note: The ST author should fill in the assignment with all attributes that the administrator is able to specify when creating the router rules.

for non-IP-based network stacks: [assignment: information security attributes]].

FDP\_IFF.1.2(2) - **Refinement:** The TSF shall permit an information flow between a **source** subject and a **destination subject** via a controlled operation if the following rules hold:

- [the source subject has successfully authenticated to the TOE;
- the identity of the destination subject is in the set of destination identifiers;
- the information security attributes match the attributes in a information flow policy rule (contained in the information flow policy ruleset defined by the administrator) according to the following algorithm [assignment: algorithm used by the TOE to match information security attributes to information flow policy rules]; and
- the selected information flow policy rule specifies that the information flow is to be permitted].

102 Application Note: In a router, the administrator specifies information flow policy rules that contain information security attribute values (or wildcards that “stand” for multiple values of the same type; e.g., 127.\*.\*.\* would represent any IP address that begins with “127”), and associated with that rule an action that permits the information flow or disallows the information flow. When a packet arrives at the source interface, the information security attribute values of the packet are

compared to each information flow policy rule by some TOE-specified algorithm, and when a match is found the action specified by that rule is taken. Since wildcards would allow the specific attributes in a packet to potentially match more than one rule, the ST author needs to fill in the assignment with the algorithm the TOE uses to find a matching rule. This could be “first match”, “most specific match”, or some more elaborate description.

FDP\_IFF.1.3(2) - The TSF shall enforce the [assignment: additional information flow control SFP rules].

FDP\_IFF.1.4(2) - The TSF shall provide the following

- [the Security Administrator shall have the capability to view all information flows allowed by the information flow policy ruleset before the ruleset is applied].

103 Application Note: Some routers create additional rules as a side-effect of creating a rule; for example, a router may create a rule allowing an FTP data channel when a rule allowing FTP (control connections) is created. This requirement allows an administrator to view the entire ruleset so that they can identify such rules and confirm that the ruleset reflects the desired policy.

104 “before the ruleset is applied” means that the administrator is able to view the entire rule set before it is put into use on the TOE. This gives the administrator the opportunity to address any errors or unintended flows.

FDP\_IFF.1.5(2) - The TSF shall explicitly authorize an information flow based on the following rules: [none].

FDP\_IFF.1.6(2) - The TSF shall explicitly deny an information flow based on the following rules: [none].

105 Application Note: Note that the checks done in FDP\_IFF.1.6(1) do not need to be re-specified because those checks should occur prior to the user authenticating themselves to the router, and thus are part of the unauthenticated policy rather than the authenticated policy.

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

FDP\_RIP.2.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the [selection: allocation of the resource to, deallocation of the resource from] all objects.

#### 5.1.4 Identification and Authentication (FIA)

TOE security functions implemented by a probabilistic or permutational mechanism (e.g., password or hash function) are required (at EAL2 and higher).

##### 5.1.4.1 Authentication failure handling (FIA\_AFL.1-NIAP-0425)

FIA\_AFL.1.1-NIAP-0425 - **Refinement:** The TSF shall detect when [a **Security Administrator**-configurable integer] of unsuccessful authentication attempts occurs related to [a user's authentication].

FIA\_AFL.1.2-NIAP-0425 – **Refinement:** When the defined number of unsuccessful authentication attempts has been met, the TSF shall [at the option of the **Security Administrator** prevent the [users except the local security administrator] from performing activities that require authentication until an action is taken by the **Security Administrator**, or until a Security Administrator defined time period has elapsed].

106 *Application Note: This requirement does not apply to the local administrators, since it does not make sense to lock a local administrator's account in this fashion. This could be addressed by requiring a separate account for local administrators, which would be stated in the administrative guidance, or the TOE's authentication mechanism implementation could distinguish login attempts that are made locally and remotely.*

##### 5.1.4.2 User attribute definition (Human User Identity) (FIA\_ATD.1(1))

FIA\_ATD.1.1(1) – **Refinement:** The TSF shall maintain the following list of security attributes belonging to an **authorized** user:

a) [user identifier(s):

role;

[selection: [assignment: Any security attributes related to a user identifier (e.g., certificate associated with the userid)], none]; and

b) [selection: [assignment: other user security attributes], none]].

107 *Application Note: This requirement applies to authorized users: administrators and authorized IT entities. The intent is to allow multiple userids to be associated with a user. This allows a single human user to assume multiple roles, albeit requiring authentication as the userid associated with a given role. The intent is for a userid to only be associated with a single role, thus limiting the amount of damage if an administrative role is compromised.*

108 If a particular TOE has different attributes for administrators and authenticated IT entities, the ST author is expected to iterate this requirement once for each set of users to reflect the differences.

109 Item “b” could be used by an ST author to specify a list of the session establishment criteria identified in FTA\_TSE depending on the TOE’s implementation of the session establishment function.

#### 5.1.4.3 User Attribute definition (TOE to TOE Identification) (FIA\_ATD.1(2))

FIA\_ATD.1.1(2) **Refinement:** The TSF shall maintain the following list of security attributes belonging to **authorized subjects**:

- a) [subject identity;
- b) [assignment: any other security attributes].

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

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

#### 5.1.4.5 Authentication Mechanism (FIA\_UAU\_(EXT).5)

FIA\_UAU\_(EXT).5.1 - The TSF shall provide a local authentication mechanism, [selection: [assignment: other authentication mechanism(s)], none] to perform user authentication.

110 Application Note: This extended requirement is needed because there is no CC requirement (other than FIA\_UAU.5) that requires the TSF provide authentication (it is implied by other FIA\_UAU requirements, but not explicitly required).

111 The ST author could chose to fill in the assignment with any additional authentication mechanism such as a single-use authentication mechanism, or a mechanism that authenticates users by using a certificate. If an asymmetric algorithm is chosen, the TOE may rely upon a certificate authority server to obtain a user’s certificate, and this server would be considered an authorized IT entity and IT environment requirements should be levied on this IT entity.

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

FIA\_UID.2.1 The TSF shall require each user to identify itself before allowing any other TSFmediated actions on behalf of that user.

#### 5.1.4.7 User-subject binding (FIA\_USB.1)

FIA\_USB.1.1: The TSF shall associate the following user security attributes with subjects acting on the behalf of that user: [assignment: list of user security attributes].

FIA\_USB.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: rules for the initial association of attributes].

FIA\_USB.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: rules for the changing of attributes].

112 Application Note: User security attributes are defined in FIA\_ATD.1

113 Interp note: This requirement is modified as per CCIMB Interp #137.

### 5.1.5 Security Management (FMT)

#### 5.1.5.1 Management of security functions behavior (TSF non-Cryptographic Self-test) (FMT\_MOF.1(1))

FMT\_MOF.1.1(1) - The TSF shall restrict the ability to modify the behavior of the functions

[TSF Self-Test (FPT\_TST\_(EXT).1)]

to [the Security Administrator].

114 Application Note: “Modify the behavior” refers to specifying the interval at which the test periodically run, or perhaps selecting a subset of the tests to run.

#### 5.1.5.2 Management of security functions behavior (Cryptographic Self-test) (FMT\_MOF.1(2))

FMT\_MOF.1.1(2) - The TSF shall restrict the ability to enable, disable the functions

[TSF Self-Test (FPT\_TST.1(1) and FPT\_TST.1(2)]

to [the Cryptographic Administrator].

115 Application Note: The enabling or disabling of the cryptographic self-tests immediately after key generation.

### 5.1.5.3 Management of security functions behavior (Audit review) (FMT\_MOF.1(3))

FMT\_MOF.1.1(3) - The TSF shall restrict the ability to enable, disable, determine and modify the behavior of the functions

[Security Audit (FAU\_SAR.1, FAU\_SAR.2, FAU\_SAR.3)] to [an Audit Administrator].

### 5.1.5.4 Management of security functions behavior (Audit selection) (FMT\_MOF.1(4))

FMT\_MOF.1.1(4) - The TSF shall restrict the ability to enable, disable, determine and modify the behavior of the functions

[Security Audit Analysis (FAU\_SAA); and

Security Audit (FAU\_SEL)]

to [the Security Administrator].

116 Application Note: For the Audit function, enable and disable refer to the ability to enable or disable the audit mechanism as a whole. “Determine the behavior” means the ability to determine specifically what on the system is being audited, while “modify the behavior” means the ability to set or unset specific aspects of the audit mechanism, such as what user behavior is audited, etc.

### 5.1.5.5 Management of security functions behavior (Alarms) (FMT\_MOF.1(5))

FMT\_MOF.1.1(5) - The TSF shall restrict the ability to enable, or disable the functions

[Security Alarms (FAU\_ARP)]

to [the Security Administrator].

117 Application Note: This requirement ensures only the Security Administrator can enable or disable (turn on or turn off) the alarm notification function – messages and/or the audible alarm. As currently written, FAU\_ARP.1 does not lend itself to behavior modification. If the ST author were to include additional functionality in FAU\_ARP.1 (e.g., notify the administrator via a pager) then the ST author should consider adding, “modify the behavior” to this requirement.

#### 5.1.5.6 Management of security functions behavior (quota mechanism) (FMT\_MOF.1(6))

FMT\_MOF.1.1(6) - The TSF shall restrict the ability to determine the behavior of the functions

[Controlled connection-oriented resource allocation (FRU\_RSA.1(2));

an administrator-specified network identifier;

set of administrator-specified network identifiers;

administrator-specified period of time]

to [the Security Administrator].

- 118 Application Note: “determine the behavior of” refers to specifying the network identifier(s) that will be tracked using the FRU\_RSA.1(2) requirement and the time period over which the quota limitations are enforced. Note that the specification of the actual quotas, while part of the resource allocation functionality, is done by FMT\_MTD.2(2).

#### 5.1.5.7 Management of security functions behavior (Authentication Attempts) (FMT\_MOF.1(7))

FMT\_MOF.1.1(7) - The TSF shall restrict the ability to *enable, disable, determine and modify the behavior* of the functions

[Authentication failure handling (FIA\_AFL.1.2) configurable integer of unsuccessful authentication attempts that occurs related to a user’s authentication to [the Security Administrator].

#### 5.1.5.8 Management of security attributes (unauthenticated) (FMT\_MSA.1(1))

FMT\_MSA.1.1 The TSF shall enforce the [UNAUTHENTICATED INFORMATION FLOW SFP] to restrict the ability to [selection: change\_default, query, modify, delete, [assignment: other operations]] the security attributes [settable percentage of storage capacity [assignment: list of security attributes]] to [Security Administrator [assignment: the authorized identified roles]].

#### 5.1.5.9 Management of security attributes (authenticated) (FMT\_MSA.1(2))

FMT\_MSA.1.1 The TSF shall enforce the [AUTHENTICATED INFORMATION FLOW SFP] to restrict the ability to [selection: change\_default, query, modify, delete, [assignment: other operations]] the security attributes [settable percentage of storage capacity [assignment: list of security attributes]] to [Security Administrator [assignment: the authorized identified roles]].

#### 5.1.5.10 Secure security attributes (FMT\_MSA.2)

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

#### 5.1.5.11 FMT\_MSA.3(1)

FMT\_MSA.3.1(1) –The TSF shall enforce the [UNAUTHENTICATED INFORMATION FLOW SFP] to provide *restrictive* default values for security attributes that are used to enforce the SFP.

119 Application Note: “restrictive” in this case means that by default information is not allowed to flow (according to the referenced policies) unless an explicit enforcing attribute allows an information flow. By default, information is not allowed to flow.

FMT\_MSA.3.2(1) - The TSF shall allow the [Security Administrator] to specify alternative initial values to override the default values when an object or information is created.

#### 5.1.5.12 FMT\_MSA.3(2)

FMT\_MSA.3.1(2) – The TSF shall enforce the [AUTHENTICATED INFORMATION FLOW SFP] to provide *restrictive* default values for security attributes that are used to enforce the SFP.

120 Application Note: “restrictive” in this case means that by default information is not allowed to flow (according to the referenced policies) unless an explicit enforcing attribute allows an information flow. By default, information is not allowed to flow.

FMT\_MSA.3.2(2) - The TSF shall allow the [Security Administrator] to specify alternative initial values to override the default values when an object or information is created.

5.1.5.13 Management of TSF data (non-cryptographic, non-time TSF data) (FMT\_MTD.1(1))

FMT\_MTD.1.1(1) – **Refinement:** The TSF shall restrict the ability to [selection: change default, query, modify, delete, clear, [**selection:** [assignment: other operations], **none**]] **all** the [TSF data except cryptographic security data and the time and date used to form the time stamps in FPT\_STM.1] to [the administrators or authorized IT entities].

121 Application Note: The ST author should iterate this requirement as necessary to ensure that the TSF data are characterized in terms of the functionality provided by the TOE, and that the access is appropriately restricted to the appropriate administrators and authorized IT entities. The cryptographic security data and time stamp data are covered in the following two components, as they have specific requirements to address the PP's threats and policies.

5.1.5.14 Management of TSF data (cryptographic TSF data) (FMT\_MTD.1(2))

FMT\_MTD.1.1(2) - The TSF shall restrict the ability to *modify* the [cryptographic security data] to [the Cryptographic Administrator].

122 Application Note: The intent of this requirement is to restrict the ability to configure the TOE's cryptographic policy to the Cryptographic Administrator. Configuring the cryptographic policy is related to things such as: setting modes of operation, key lifetimes, selecting a specific algorithm, and key length.

5.1.5.15 Management of TSF data (time TSF data) (FMT\_MTD.1(3))

FMT\_MTD.1.1(3) – The TSF shall restrict the ability to [set] the [time and date used to form the time stamps in FPT\_STM.1] to [the Security Administrator or authorized IT entity].

123 Application Note: The ST author is able to restrict the ability to set the time and date to just the Security Administrator, or just an authorized IT entity, or both.

5.1.5.16 Management of TSF data (Information flow policy ruleset) (FMT\_MTD.1(4))

FMT\_MTD.1.1(4) – The TSF shall restrict the ability to *query, modify, delete, create,* [selection: [assignment: other operations], none] the [information flow policy rules] to [the Security Administrator].

- 124 Application Note: This restricts the specification of the information flow policy ruleset identified in the FDP\_IFF requirements to the Security Administrator. This specification is done using the attributes defined for those policies.
- 125 The ST author should fill in any TOE-specific operations that an administrator can perform on the ruleset in the assignment.

#### 5.1.5.17 Management of limits on TSF data (transport-layer quotas) (FMT\_MTD.2(1))

FMT\_MTD.2.1(1) - The TSF shall restrict the specification of the limits for [quotas on transport-layer connections] to [the Security Administrator].

FMT\_MTD.2.2(1) - The TSF shall take the following actions, if the TSF data are at, or exceed, the indicated limits: [assignment: actions to be taken].

- 126 Application Note: Note that the wording of FRU\_RSA.1(1) does not indicate that the TOE must provide the Security Administrator the means to adjust the maximum quota; however, if the TOE does provide such a mechanism then FMT\_MTD.2.1(1) would require that that mechanism is restricted to the Security Administrator.

- 127 For FMT\_MTD.2.2(1), the ST author should specify the actions that the TOE takes when quota is reached. For the TCP SYN attack, for example, the action may be to drop the oldest “n” half-open connections.

#### 5.1.5.18 Management of limits on TSF data (controlled connection-oriented quotas) (FMT\_MTD.2(2))

FMT\_MTD.2.1(2) - The TSF shall restrict the specification of the limits for [quotas on controlled connection-oriented resources] to [the Security Administrator].

FMT\_MTD.2.2(2) - The TSF shall take the following actions, if the TSF data are at, or exceed, the indicated limits: [assignment: actions to be taken].

- 128 Application Note: For FMT\_MTD.2.2(2), the ST author should specify the actions that the TOE takes for each controlled connection-oriented resource when the quota (with respect the specific network identifier or set of network identifiers) established by the Security Administrator is reached. This requirement may have to be iterated to be consistent with FRU\_RSA.1(2). See the application note on FRU\_RSA.1(2) for more detail on the requirements for the quota mechanism.

#### 5.1.5.19 Management of limits on TSF data (percentage of storage capacity) (FMT\_MTD.2(3))

FMT\_MTD.2.1(3) - The TSF shall restrict the specification of the limits for [percentage of storage capacity for audit records] to [the Security Administrator].

FMT\_MTD.2.2(3) - The TSF shall take the following actions, if the TSF data are at, or exceed, the indicated limits: [assignment: actions to be taken].

129 For FMT\_MTD.2.2(1), the ST author should specify the actions that the TOE takes when percentage is reached. This assignment should be in concert with the selection/assignment performed in FAU\_STG.NIAP-0414-1.1-NIAP-0429.

#### 5.1.5.20 Revocation (FMT\_REV.1)

FMT\_REV.1.1 The TSF shall restrict the ability to revoke security attributes associated with the [selection: users, subjects, objects, [assignment: other additional resources]] within the TSC to [Security administrator].

FMT\_REV.1.2 The TSF shall enforce the rules [assignment: specification of revocation rules].

130 Interp note: This requirement is modified as per CCIMB Interp #201

#### 5.1.5.21 Specification of Management Functions (FMT\_SMF.1)

FMT\_SMF.1.1 The TSF shall be capable of performing the following security management functions: [

1. restrict the ability to invoke determine and modify the behavior of the functions TSF Self-Test (FPT\_TST\_(EXT).1) to the Security Administrator;
2. restrict the ability to enable, disable the functions TSF Self-Test (FPT\_TST.1(1) and FPT\_TST.1(2)) to the Cryptographic Administrator;
3. restrict the ability to enable, disable, determine and modify the behavior of the functions Security Audit (FAU\_SAR.1, FAU\_SAR.2, FAU\_SAR.3) to an Audit Administrator;
4. restrict the ability to enable, disable, determine and modify the behavior of the functions Security Audit Analysis (FAU\_SAA); and Security Audit (FAU\_SEL) to the Security Administrator;

5. restrict the ability to enable, or disable the functions Security Alarms (FAU\_ARP) to the Security Administrator
6. restrict the ability to determine the behavior of the functions: Controlled connection-oriented resource allocation (FRU\_RSA.1(2)); an administrator-specified network identifier; set of administrator-specified network identifiers; administrator-specified period of time to the Security Administrator.
7. enforce administrator-specified maximum quotas of the following resources: [controlled connection-oriented resources] that users associated with [an administrator-specified network identifier and a set of administrator-specified network identifiers] can use over an administrator-specified period of time.
8. enforce the [UNAUTHENTICATED INFORMATION FLOW SFP] to provide restrictive default values security attributes that are used to enforce the SFP;
9. enforce the [AUTHENTICATED INFORMATION FLOW SFP] to provide restrictive default values security attributes that are used to enforce the SFP;
10. restrict the ability to [selection: change default, query, modify, delete, clear, [selection: [assignment: other operations], none]] all the [TSF data except cryptographic security data and the time and date used to form the time stamps in FPT\_STM.1] to the administrators or authorized IT entities (FMT\_MTD.1(1)).
11. restrict the ability to modify the cryptographic security data to the Cryptographic Administrator (FMT\_MTD.1(2));
12. restrict the ability to set the time and date used to form the time stamps in FPT\_STM.1 to the Security Administrator or authorized IT entity.
13. restrict the ability to query, modify, delete, create, [selection: [assignment: other operations], none] the information flow policy rules to the Security Administrator (FMT\_MTD.1(2)).
14. restrict the ability to set the time and date used to form the time stamps in FPT\_STM.1 to [the Security Administrator or authorized IT entity (FMT\_MTD.1.1(3))].
15. restrict the ability to *query, modify, delete, create*, [selection: [assignment: other operations], none] the [information flow policy rules] to [the Security Administrator] (FMT\_MTD.1.1(4)).

16. restrict the ability to revoke security attributes associated with the [selection: users, subjects, objects, [assignment: other additional resources]] within the TSC to Security administrator (FMT\_REV.1)
17. restrict the specification of the limits for quotas on transport-layer connections to the Security Administrator (FMT\_MTD.2 (1));
18. restrict the specification of the limits for quotas on controlled connection-oriented resources to the Security Administrator (FMT\_MTD.2 (2));
19. restrict the specification of the limits for percentage of storage capacity for audit records (FMT\_MTD.2(3)) to the Security Administrator.
20. restrict the ability to enable, disable, determine and modify the behavior of the functions of Authentication failure handling (FIA\_AFL.1.2) to configure an integer of unsuccessful authentication attempts that occurs related to a user's authentication to the Security Administrator.
21. [assignment: list of additional security management functions to be provided by the IT environment]].

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

FMT\_SMR.2.1 The TSF shall maintain the roles: [

- Security administrator role,
- Audit administrator role,
- Crypto administrator role].

FMT\_SMR.2.2 The TSF shall be able to associate users with roles.

FMT\_SMR.2.3 The TSF shall ensure that the conditions [assignment: conditions for the different roles] are satisfied.

131 Application Note: The administering of the TOE is limited to the capabilities associated with an administrative role.

### 5.1.6 Protection of the TSF (FPT)

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

FPT\_FLS.1.1 The TSF shall preserve a secure state when the following types of failures occur: [all failures].

#### 5.1.6.2 Inter-TSF availability within a defined availability metric (FPT\_ITA.1)

FPT\_ITA.1.1 The TSF shall ensure the availability of [assignment: list of types of TSF data] provided to a remote trusted IT product within [assignment: a defined availability metric] given the following conditions [assignment: conditions to ensure availability].

- 132 Application Note: The router must be sure to follow specifications for the different routing protocols used (e.g., BGP, OSPF, etc.) for the retransmission and availability of routing data to peer routers.

#### 5.1.6.3 Inter-TSF confidentiality during transmission (FPT\_ITC.1)

FPT\_ITC.1.1 The TSF shall protect all TSF data transmitted from the TSF to a remote trusted IT product from unauthorized disclosure during transmission.

- 133 Application Note: This is to protect routing and routing related updates between two routers. An example would be the use of Internet Protocol Version 6 (IPv6) and Internet Protocol Security (IPSEC) so that a router could encrypt routing table communications and authentication with a peer router. This would be a method to protect the confidentiality of the network resources.

#### 5.1.6.4 Inter-TSF detection of modification (FPT\_ITI.1)

FPT\_ITI.1.1 The TSF shall provide the capability to detect modification of all TSF data during transmission between the TSF and a remote trusted IT product within the following metric: [assignment: a defined modification metric].

FPT\_ITI.1.2 The TSF shall provide the capability to verify the integrity of all TSF data transmitted between the TSF and a remote trusted IT product and perform [assignment: action to be taken] if modifications are detected.

- 134 Application Note: This is to protect routing and routing related updates between two routers. An example would be the use of Internet Protocol Version 6 (IPv6) and Internet Protocol Security (IPSEC) so that a router could protect and detect modification of peer communications by encrypting routing table communication and authentication with a peer router.

#### 5.1.6.5 Standard Protocol Usage (FPT\_PRO\_(EXT).1)

FPT\_PRO\_(EXT).1 The TSF shall utilize the standard protocol mechanisms within the standard protocols

[selection:[assignment:

- a) BGP
- b) list of standard protocols]].

#### 5.1.6.6 Automated Recovery (FPT\_RCV.2)

FPT\_RCV.2.1 When automated recovery from [assignment: list of failures/service discontinuities] is not possible, the TSF shall enter a maintenance mode where the ability to return to a secure state is provided.

FPT\_RCV.2.2 For [assignment: list of failures/service discontinuities], the TSF shall ensure the return of the TOE to a secure state using automated procedures.

#### 5.1.6.7 Replay detection (FPT\_RPL.1)

FPT\_RPL.1.1 - The TSF shall detect replay for the following entities: [TSF data and security attributes].

FPT\_RPL.1.2 - The TSF shall perform

[reject data;

audit event; and

[selection: [assignment: list of specific actions], none]]

when replay is detected.

135 Application Note: Receiving multiple network packets due to network congestion or lost packet acknowledgments is not considered a replay attack. The intent of this requirement is to ensure that an administrative session (in part, in its entirety, by a remote administrator or an authorized IT entity) or a user's authentication sequence cannot be replayed.

#### 5.1.6.8 Reliable time stamps (FPT\_STM.1)

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

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

FPT\_TDC.1.1 The TSF shall provide the capability to consistently interpret [assignment: list of TSF data types] when shared between the TSF and another trusted IT product.

FPT\_TDC.1.2 The TSF shall use [assignment: list of interpretation rules to be applied by the TSF] when interpreting the TSF data from another trusted IT product.

#### 5.1.6.10 Extended: TSF Testing (FPT\_TST\_EXP.1)

FPT\_TST\_EXP.1.1 The TSF shall run a suite of self tests during the initial start-up and also either periodically during normal operation, or at the request of an authorized administrator to demonstrate the correct operation of the TSF.

FPT\_TST\_EXP.1.2 The TSF shall provide authorized administrators with the capability to verify the integrity of stored TSF executable code through the use of the TSF-provided cryptographic services.

*Application Note: Refer to FCS\_COP.1.1(2) and FCS\_COP.1.1(3) for TSF-provided cryptographic services .*

#### 5.1.6.11 TSF Testing (for cryptography) (FPT\_TST.1(1))

FPT\_TST.1.1(1) **Refinement:** The TSF shall run a suite of self tests **in accordance with FIPS PUB 140-2 and Appendix C of this profile** during initial start-up (on power on), at the request of the cryptographic administrator (on demand), under various conditions defined in section 4.9.1 of FIPS 140-2, and periodically (at least once a day) to demonstrate the correct operation of the **following cryptographic functions:**

- a) **key error detection;**
- b) **cryptographic algorithms;**
- c) **RNG/PRNG**

*Application Note: These tests apply regardless of whether the cryptographic functionality is implemented in hardware, software, or firmware.*

FPT\_TST.1.2(1) **Refinement:** The TSF shall provide authorized **cryptographic administrators** with the capability to verify the integrity of **TSF data related to the cryptography by using TSF-provided cryptographic functions.**

*Application Note: Refer to FCS\_COP.1.1(2) and FCS\_COP.1.1(3) for TSF-provided cryptographic services*

FPT\_TST.1.3(1) **Refinement:** The TSF shall provide authorized **cryptographic administrators** with the capability to verify the integrity of stored TSF executable code **related to the**

**cryptography by using TSF-provided cryptographic functions.iii**

*Application Note: Refer to FCS\_COP.1.1(2) and FCS\_COP.1.1(3) for TSF-provided cryptographic services .*

**5.1.6.12 TSF Testing (for key generation components) (FPT\_TST.1(2))**

**FPT\_TST.1.1(2) Refinement:** The TSF shall **perform self tests immediately after generation of a key** to demonstrate the correct operation of each key generation component. **If any of these tests fails, that generated key shall not be used, the cryptographic module shall react as required by FIPS PUB 140-2 for failing a self-test, and this event will be audited.iv**

*Application Note: Key generation components are those critical elements that compose the entire key generation process (e.g., any algorithms, any RNG/PRNGs, any key generation seeding processes, etc.).*

*Application Note: These self-tests on the key generation components can be executed here as a subset of the full suite of self-tests run on the cryptography in FPT\_TST.1(1) as long as all elements of the key generation process are tested.*

**FPT\_TST.1.2(2) Refinement:** The TSF shall provide authorized **cryptographic administrators** with the capability to verify the integrity of TSF data **related to the key generation by using TSF-provided cryptographic functions.v**

*Application Note: Refer to FCS\_COP.1.1(2) and FCS\_COP.1.1(3) for TSF-provided cryptographic services*

**FPT\_TST.1.3(2) Refinement:** The TSF shall provide authorized **cryptographic administrators** with the capability to verify the integrity of stored TSF executable code **related to the key generation by using TSF-provided cryptographic functions.vi**

*Application Note: Refer to FCS\_COP.1.1(2) and FCS\_COP.1.1(3) for TSF-provided cryptographic services .*

**5.1.7 Resource Utilization (FRU)**

**5.1.7.1 Maximum quotas (transport-layer quotas) (FRU\_RSA.1(1))**

**FRU\_RSA.1.1(1) – Refinement:** The TSF shall enforce maximum quotas of the following resources: [transport-layer representation] that ***a source subject identifier*** can use ***over a specified period of time.***

136 Application Note: “transport-layer representation” refers specifically to the TCP SYN attack, where half-open connections are established thus exhausting the connection table resource. The selection for this requirement was refined to specify a source subject identifier, which is more accurate than user or subject in the context of a router. If the TOE does not implement the TCP/IP protocol, this

requirement would apply to a similar type of transport-layer entity for that TOE's protocol stack.

#### 5.1.7.2 Maximum quotas (controlled connection-oriented quotas) (FRU\_RSA.1(2))

FRU\_RSA.1.1(2) – **Refinement:** The TSF shall enforce **administrator-specified** maximum quotas of the following resources: [controlled connection-oriented resources] that **users associated with** [*an administrator-specified network identifier and a set of administrator-specified network identifiers*] can use *over an administrator-specified period of time*.

137 Application Note: This requirement applies to a network entity attempting to exhaust the specified connection-oriented resources (or set of such resources) on the TOE. Connectionless sessions are not a concern because they do not consume resources that persist like connection-oriented sessions do.

138 The ST author should fill in the first assignment with the list of connection-oriented resources to which this requirement applies. That is, when a network entity uses such a connection-oriented resource (or a collection of these resources), the TOE tracks that use for the purpose of determining whether the entity has exceed the quota established by the administrator.

139 The ST author should use the first selection to indicate whether the TOE is able to track the assignment of the specified resources based on a single network identifier (e.g., a specific IP address) or multiple network identifiers (e.g., a specific IP subnet address). The second selection should reflect the way in which the TOE tracks such resource use. Note that the ST author may have to iterate this requirement if different resources can be controlled differently by the TOE. The ST author should ensure that FMT\_MTD.2(2) specifies the actions that are taken for each resource on which there is a quota.

### 5.1.8 TOE Access (FTA)

#### 5.1.8.1 TSF-initiated termination (FTA\_SSL.3)

FTA\_SSL.3.1 The TSF shall terminate an interactive session after a [assignment: time interval of user inactivity].

#### 5.1.8.2 Default TOE access banners (FTA\_TAB.1)

FTA\_TAB.1.1 **Refinement:** Before establishing a user session **that requires authentication**, the TSF shall display **only a Security Administrator specified**

advisory **notice and consent** warning message regarding unauthorized use of the TOE.

- 140 Application Note: The access banner applies whenever the TOE will provide a prompt for identification and authentication. The intent of this requirement is to advise users of warnings regarding the unauthorized use of the TOE and to provide the Security Administrator with control over what is displayed (e.g., if the Security Administrator chooses, they can remove banner information that informs the user of the product and version number).

### 5.1.8.3 TOE session establishment (FTA\_TSE.1)

FTA\_TSE.1.1 The TSF shall be able to deny session establishment based on [assignment: attributes].

## 5.1.9 Trusted Path/Channels (FTP)

### 5.1.9.1 Inter-TSF trusted channel (Prevention of Disclosure) (FTP\_ITC.1(1))

FTP\_ITC.1.1(1) - **Refinement:** The TSF shall **use encryption to provide a trusted communication channel between itself and a remote-trusted authorized IT product-entities** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from ~~modifications or disclosure~~.

FTP\_ITC.1.2(1) **Refinement:** The TSF shall permit *the TSF, or the remote-trusted authorized IT product-entities* to initiate communication via the trusted channel.

- 141 Application Note: The encryption used to protect the communication channel from disclosure is one of the symmetric algorithms specified in FCS\_COP.1(1).

- 142 FTP\_ITC.1.2 is used to ensure secure communications between the TOE and authorized IT entities (e.g., peer router). While these authorized IT entities may initiate communications, it may be the case that the TOE is required to perform a “pull” operation (e.g., obtaining routing information from a peer router).

FTP\_ITC.1.3(1) - The TSF shall initiate communication via the trusted channel for [all authentication functions, [selection: [assignment: list of other functions for which a trusted channel is required], none]].

- 143 Application Note: The “other functions” are the services that are provided by the authorized IT entities (e.g., RIP).

### 5.1.9.2 Inter-TSF trusted channel (Detection of Modification) (FTP\_ITC.1(2))

FTP\_ITC.1.1(2) - **Refinement:** The TSF shall **use a cryptographic signature** to provide a **trusted** communication channel between itself and **authorized IT product entities** that is logically distinct from other communication channels and provides assured identification of its end points and detection **of the modification or disclosure of data**.

FTP\_ITC.1.2(2) - **Refinement:** The TSF shall permit the *TSF*, **or the remote-trusted authorized IT product entities** to initiate communication via the trusted channel.

144 Application Note: The method used to provide detection of data modification transmitted through the communication channel is the cryptographic digital signature algorithm specified in FCS\_COP.1(2).

145 FTP\_ITC.1.2 is used to ensure secure communications between the TOE and authorized IT entities (e.g., peer router). While these authorized IT entities may initiate communications, it may be the case that the TOE is required to perform a “pull” operation (e.g., obtaining routing information from a peer router).

FTP\_ITC.1.3(2) - The TSF shall initiate communication via the trusted channel for [all authentication functions, [selection: [assignment: list of other functions for which a trusted channel is required], none]].

146 Application Note: The “other functions” are the services that are provided by the authorized IT entities (e.g., RIP).

### 5.1.9.3 Trusted path (Prevention of Disclosure) (FTP\_TRP.1(1))

FTP\_TRP.1.1(1) - **Refinement:** The TSF shall provide **a an encrypted** communication path between itself and *remote administrators and authenticated* users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from disclosure.

FTP\_TRP.1.2(1) - The TSF shall permit *remote users* to initiate communication via the trusted path.

FTP\_TRP.1.3(1) – **Refinement:** The TSF shall require the use of the trusted path for *initial user authentication*, all remote administration actions, [selection: [assignment: other services for which trusted path is required, none]].

147 Application Note: The encryption used to protect the communication channel from disclosure is one of the symmetric algorithms specified in FCS\_COP.1(1)

148 “all remote administration actions” means that the entire remote administration session is protected with the trusted path; that is, the administrator is assured of communicating with the TOE and the data passing between the administrator and the TOE are protected from disclosure.

#### 5.1.9.4 Trusted path (Detection of Modification) (FTP\_TRP.1(2))

FTP\_TRP.1.1(2) - **Refinement:** The TSF shall **use a cryptographic signature to** provide a communication path between itself and *remote administrators and authenticated* users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from ~~modification or disclosure~~ **detection of the modification of data.**

FTP\_TRP.1.2(2) - The TSF shall permit *remote users* to initiate communication via the trusted path.

FTP\_TRP.1.3(2) – **Refinement:** The TSF shall require the use of the trusted path for *initial user authentication*, all remote administration actions, [selection: [assignment: other services for which trusted path is required, none]].

149 Application Note: The method used to provide detection of data modification transmitted through the communication channel is the cryptographic digital signature algorithm specified in FCS\_COP.1(2).

150 “all remote administration actions” means that the entire remote administration session is protected with the trusted path; that is, the administrator is assured of communicating with the TOE and the data passing between the administrator and the TOE provides a means for detecting the modification of data that flows through the protected communication path.

## 5.2 Security Requirements for the IT Environment

151 This PP does not require any security requirements on the IT environment; therefore no additional rationale is needed here.

## 5.3 TOE Security Assurance Requirements

This section defines the assurance requirements for the TOE. Table 5 summarizes the components for medium robustness. The augmented requirements are in bold print.

The TOE assurance requirements for this PP do not map to a CC EAL. The assurance requirements are summarized in the Table below, with the extended requirements in bold print..

**Table 5 Assurance Requirements**

Assurance Class	Assurance Components	Assurance Components Description
-----------------	----------------------	----------------------------------

## Router Protection Profile For Medium Robustness Environments

Assurance Class	Assurance Components	Assurance Components Description
Development	ADV_ARC.1	Security Architectural Description
	<b>ADV_FSP.5</b>	<b>Complete semi-formal functional specification with additional error information</b>
	ADV_IMP.1	Implementation of the TSF
	<b>ADV_INT.3</b>	<b>Minimally complex internals</b>
	<b>ADV_TDS.4</b>	<b>Semiformal modular design</b>
Guidance Documents	AGD_OPE.1	Operational user guidance
	AGD_PRE.1	Preparative User guidance
Life Cycle Support	ALC_CMC.4	Product support, acceptance procedures and automation
	ALC_CMS.4	Problem tracking CM coverage
	ALC_DEL.1	Delivery procedures
	ALC_DVS.1	Identification of security measures
	<b>ALC_FLR.2</b>	<b>Flaw Reporting Procedures</b>
	ALC_LCD.1	Developer defined life-cycle model
	ALC_TAT.1	Well-defined development tools
Tests	ATE_COV.2	Analysis of coverage
	<b>ATE_DPT.3</b>	<b>Testing: modular design</b>
	ATE_FUN.1	Functional testing
	ATE_IND.2	Independent testing - sample
Vulnerability Assessment	<b>AVA_CCA_(EXT).1</b>	<b>Systematic cryptographic module covert channel analysis (required when Cryptography is invoked)</b>
	<b>AVA_VAN.4</b>	<b>Methodical vulnerability analysis</b>

## 5.3.1 Class ADV: Development

### 5.3.1.1 ADV\_ARC.1 Security architecture description

Dependencies: ADV\_FSP.1 Basic functional specification

ADV\_TDS.1 Basic design

Developer action elements:

ADV\_ARC.1.1D The developer shall design and implement the TOE so that the security features of the TSF cannot be bypassed.

ADV\_ARC.1.2D The developer shall design and implement the TSF so that it is able to protect itself from tampering by untrusted active entities.

ADV\_ARC.1.3D The developer shall provide a security architecture description of the TSF.

Content and presentation elements:

ADV\_ARC.1.1C The security architecture description shall be at a level of detail commensurate with the description of the SFR-enforcing abstractions described in the TOE design document.

ADV\_ARC.1.2C The security architecture description shall describe the security domains maintained by the TSF consistently with the SFRs.

ADV\_ARC.1.3C The security architecture description shall describe how the TSF initialization process is secure.

ADV\_ARC.1.4C The security architecture description shall demonstrate that the TSF protects itself from tampering.

ADV\_ARC.1.5C The security architecture description shall demonstrate that the TSF prevents bypass of the SFR-enforcing functionality.

Evaluator action elements:

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

### 5.3.1.2 ADV\_FSP.5 Complete semi-formal functional specification with additional error information

Dependencies: ADV\_TDS.1 Basic design,

ADV\_IMP.1 Implementation representation of the TSF Developer action elements:

Developer action elements:

ADV\_FSP.5.1D The developer shall provide a functional specification.

- ADV\_FSP.5.2D The developer shall provide a tracing from the functional specification to the SFRs.  
Content and presentation elements:
- ADV\_FSP.5.1C The functional specification shall completely represent the TSF.
- ADV\_FSP.5.2C The functional specification shall describe the TSFI using a semi-formal style.
- ADV\_FSP.5.3C The functional specification shall describe the purpose and method of use for all TSFI.
- ADV\_FSP.5.4C The functional specification shall identify and describe all parameters associated with each TSFI.
- ADV\_FSP.5.5C The functional specification shall describe all actions associated with each TSFI.
- ADV\_FSP.5.6C The functional specification shall describe all direct error messages that may result from an invocation of each TSFI.
- ADV\_FSP.5.7C The functional specification shall describe all error messages that do not result from an invocation of a TSFI.
- ADV\_FSP.5.8C The functional specification shall provide a rationale for each error message contained in the TSF implementation yet does not result from an invocation of a TSFI.
- ADV\_FSP.5.9C The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification.  
Evaluator action elements:
- Evaluator action elements:
- ADV\_FSP.5.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
- ADV\_FSP.5.2E The evaluator shall determine that the functional specification is an accurate and complete instantiation of the SFRs.

### **5.3.1.3 ADV\_IMP.1 Implementation representation of the TSF**

Dependencies: ADV\_TDS.3 Basic modular design  
ALC\_TAT.1 Well-defined development tools

Developer action elements:

- ADV\_IMP.1.1D The developer shall make available the implementation representation for the entire TSF.
- ADV\_IMP.1.2D The developer shall provide a mapping between the TOE design description and the sample of the implementation representation.
- Content and presentation elements:
- ADV\_IMP.1.1C The implementation representation shall define the TSF to a level of detail such that the TSF can be generated without further design decisions.

ADV\_IMP.1.2C The implementation representation shall be in the form used by the development personnel.

ADV\_IMP.1.3C The mapping between the TOE design description and the sample of the implementation representation shall demonstrate their correspondence.

Evaluator action elements:

ADV\_IMP.1.1E The evaluator shall confirm that, for the selected sample of the implementation representation, the information provided meets all requirements for content and presentation of evidence.

### ADV\_INT.3 Minimally complex internals

Dependencies: ADV\_IMP.1 Implementation representation of the TSF  
ADV\_TDS.3 Basic modular design  
ALC\_TAT.1 Well-defined development tools

Developer action elements:

ADV\_INT.3.1D The developer shall design and implement the entire TSF such that it has well-structured internals.

ADV\_INT.3.2D The developer shall provide an internals description and justification.

Content and presentation elements:

ADV\_INT.3.1C The justification shall describe the characteristics used to judge the meaning of “well-structured” and “complex”.

ADV\_INT.3.2C The TSF internals description shall demonstrate that the entire TSF is well-structured.

Evaluator action elements:

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

ADV\_INT.3.2E The evaluator *shall perform* an internals analysis on the entire TSF.

### 5.3.1.4 ADV\_TDS.4 Semiformal modular design

Dependencies: ADV\_FSP.5 Complete semi-formal functional specification with additional error information  
Developer action elements:

Developer action elements:

ADV\_TDS.4.1D The developer shall provide the design of the TOE.

ADV\_TDS.4.2D The developer shall provide a mapping from the TSFI of the functional specification to the lowest level of decomposition available in the TOE design. Content and presentation elements:

Content and presentation elements:

ADV\_TDS.4.1C The design shall describe the structure of the TOE in terms of subsystems.

ADV\_TDS.4.2C The design shall describe the TSF in terms of modules, designating each module as SFR-enforcing, SFR-supporting, or SFR-non-interfering.

ADV\_TDS.4.3C The design shall identify all subsystems of the TSF.

ADV\_TDS.4.4C The design shall provide a semiformal description of each subsystem of the TSF, supported by informal, explanatory text where appropriate.

ADV\_TDS.4.5C The design shall provide a description of the interactions among all subsystems of the TSF.

ADV\_TDS.4.6C The design shall provide a mapping from the subsystems of the TSF to the modules of the TSF.

ADV\_TDS.4.7C The design shall describe each SFR-enforcing and SFR-supporting module in terms of its purpose and interaction with other modules.

ADV\_TDS.4.8C The design shall describe each SFR-enforcing and SFR-supporting module in terms of its SFR-related interfaces, return values from those interfaces, interaction with and called interfaces to other modules.

ADV\_TDS.4.9C The design shall describe each SFR-non-interfering module in terms of its purpose and interaction with other modules.

ADV\_TDS.4.10C The mapping shall demonstrate that all behaviour described in the TOE design is mapped to the TSFIs that invoke it. Evaluator action elements:

Evaluator action elements:

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

ADV\_TDS.4.2E The evaluator shall determine that the design is an accurate and complete instantiation of all security functional requirements.

## **5.3.2 Class AGD: Guidance documents**

### **5.3.2.1 AGD\_OPE.1 Operational user guidance**

Dependencies: ADV\_FSP.1 Basic functional specification

Developer action elements:

AGD\_OPE.1.1D The developer shall provide operational user guidance.

Content and presentation elements:

- AGD\_OPE.1.1C The operational user guidance shall describe, for each user role, the user-accessible functions and privileges that should be controlled in a secure processing environment, including appropriate warnings.
- AGD\_OPE.1.2C The operational user guidance shall describe, for each user role, how to use the available interfaces provided by the TOE in a secure manner.
- AGD\_OPE.1.3C The operational user guidance shall describe, for each user role, the available functions and interfaces, in particular all security parameters under the control of the user, indicating secure values as appropriate.
- AGD\_OPE.1.4C The operational user guidance shall, for each user role, clearly present each type of security-relevant event relative to the user-accessible functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.
- AGD\_OPE.1.5C The operational user guidance shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation.
- AGD\_OPE.1.6C The operational user guidance shall, for each user role, describe the security measures to be followed in order to fulfill the security objectives for the operational environment as described in the ST.
- AGD\_OPE.1.7C The operational user guidance shall be clear and reasonable.

Evaluator action elements:

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

### **5.3.2.2 AGD\_PRE.1 Preparative procedures**

Dependencies: No dependencies.

Developer action elements:

- AGD\_PRE.1.1D The developer shall provide the TOE including its preparative procedures.

Content and presentation elements:

- AGD\_PRE.1.1C The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.
- AGD\_PRE.1.2C The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the operational environment in accordance with the security objectives for the operational environment as described in the ST.

Evaluator action elements:

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

AGD\_PRE.1.2E The evaluator shall apply the preparative procedures to confirm that the TOE can be prepared securely for operation.

### **5.3.3 Class ALC: Life-cycle support**

#### **5.3.3.1 ALC\_CMC.4 Production support, acceptance procedures and automation**

Dependencies: ALC\_CMS.1 TOE CM coverage  
ALC\_DVS.1 Identification of security measures  
ALC\_LCD.1 Developer defined life-cycle model

ALC\_CMC.4.1D The developer shall provide the TOE and a reference for the TOE.

ALC\_CMC.4.2D The developer shall provide the CM documentation.

ALC\_CMC.4.3D The developer shall use a CM system.

Content and presentation elements:

ALC\_CMC.4.1C The TOE shall be labeled with its unique reference.

ALC\_CMC.4.2C The CM documentation shall describe the method used to uniquely identify the configuration items.

ALC\_CMC.4.3C The CM system shall uniquely identify all configuration items.

ALC\_CMC.4.4C The CM system shall provide automated measures such that only authorized changes are made to the configuration items.

ALC\_CMC.4.5C The CM system shall support the production of the TOE by automated means.

ALC\_CMC.4.6C The CM documentation shall include a CM plan.

ALC\_CMC.4.7C The CM plan shall describe how the CM system is used for the development of the TOE.

ALC\_CMC.4.8C The CM plan shall describe the procedures used to accept modified or newly created configuration items as part of the TOE.

ALC\_CMC.4.9C The evidence shall demonstrate that all configuration items are being maintained under the CM system.

ALC\_CMC.4.10C The evidence shall demonstrate that the CM system is being operated in accordance with the CM plan.

Evaluator action elements:

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

### 5.3.3.2 ALC\_CMS.4 Problem tracking CM coverage

Dependencies: No dependencies.

Developer action elements:

ALC\_CMS.4.1D The developer shall provide a configuration list for the TOE.

Content and presentation elements:

ALC\_CMS.4.1C The configuration list shall include the following: the TOE itself; the evaluation evidence required by the SARs; the parts that comprise the TOE; the implementation representation; and security flaw reports and resolution status.

ALC\_CMS.4.2C The configuration list shall uniquely identify the configuration items.

ALC\_CMS.4.3C For each TSF relevant configuration item, the configuration list shall indicate the developer of the item.

Evaluator action elements:

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

### 5.3.3.3 ALC\_DEL.1 Delivery procedures

Dependencies: No dependencies.

Developer action elements:

ALC\_DEL.1.1D The developer shall document procedures for delivery of the TOE or parts of it to the consumer.

ALC\_DEL.1.2D The developer shall use the delivery procedures.

Content and presentation elements:

ALC\_DEL.1.1C The delivery documentation shall describe all procedures that are necessary to maintain security when distributing versions of the TOE to the consumer.

Evaluator action elements:

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

### 5.3.3.4 ALC\_DVS.1 Identification of security measures

Dependencies: No dependencies.

Developer action elements:

ALC\_DVS.1.1D The developer shall produce development security documentation.

Content and presentation elements:

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

Evaluator action elements:

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

ALC\_DVS.1.2E The evaluator shall confirm that the security measures are being applied.

### **5.3.3.5 ALC\_FLR.2 Flaw reporting procedures**

Dependencies: No dependencies.

Developer action elements:

ALC\_FLR.2.1D The developer shall document flaw remediation procedures addressed to TOE developers.

ALC\_FLR.2.2D The developer shall establish a procedure for accepting and acting upon all reports of security flaws and requests for corrections to those flaws.

ALC\_FLR.2.3D The developer shall provide flaw remediation guidance addressed to TOE users.

Content and presentation elements:

ALC\_FLR.2.1C The flaw remediation procedures documentation shall describe the procedures used to track all reported security flaws in each release of the TOE.

ALC\_FLR.2.2C The flaw remediation procedures shall require that a description of the nature and effect of each security flaw be provided, as well as the status of finding a correction to that flaw.

ALC\_FLR.2.3C The flaw remediation procedures shall require that corrective actions be identified for each of the security flaws.

ALC\_FLR.2.4C The flaw remediation procedures documentation shall describe the methods used to provide flaw information, corrections and guidance on corrective actions to TOE users.

ALC\_FLR.2.5C The flaw remediation procedures shall describe a means by which the developer receives from TOE users reports and enquiries of suspected security flaws in the TOE.

ALC\_FLR.2.6C The procedures for processing reported security flaws shall ensure that any reported flaws are remediated and the remediation procedures issued to TOE users.

ALC\_FLR.2.7C The procedures for processing reported security flaws shall provide safeguards that any corrections to these security flaws do not introduce any new flaws.

ALC\_FLR.2.8C The flaw remediation guidance shall describe a means by which TOE users report to the developer any suspected security flaws in the TOE.

Evaluator action elements:

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

### **5.3.3.6 ALC\_LCD.1 Developer defined life-cycle model**

Dependencies: No dependencies.

Developer action elements:

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

ALC\_LCD.1.2D The developer shall provide life-cycle definition documentation.

Content and presentation elements:

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

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

Evaluator action elements:

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

### **5.3.3.7 ALC\_TAT.1 Well-defined development tools**

Dependencies: ADV\_IMP.1 Implementation representation of the TSF

Developer action elements:

ALC\_TAT.1.1D The developer shall identify each development tool being used for the TOE.

ALC\_TAT.1.2D The developer shall document the selected implementation-dependent options of each development tool.

Content and presentation elements:

ALC\_TAT.1.1C Each development tool used for implementation shall be well-defined.

ALC\_TAT.1.2C The documentation of each development tool shall unambiguously define the meaning of all statements as well as all conventions and directives used in the implementation.

ALC\_TAT.1.3C The documentation of each development tool shall unambiguously define the meaning of all implementation-dependent options.

Evaluator action elements:

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

### **5.3.4 Class ATE: Tests**

#### **5.3.4.1 ATE\_COV.2 Analysis of coverage**

Dependencies: ADV\_FSP.2 Security-enforcing functional specification  
ATE\_FUN.1 Functional testing

Developer action elements:

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

Content and presentation elements:

ATE\_COV.2.1C The analysis of the test coverage shall demonstrate the correspondence between the tests in the test documentation and the TSFIs in the functional specification.

ATE\_COV.2.2C The analysis of the test coverage shall demonstrate that all TSFIs in the functional specification have been tested.

Evaluator action elements:

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

#### **5.3.4.2 ATE\_DPT.3 Testing: modular design**

Dependencies: ADV\_ARC.1 Security architecture description  
ADV\_TDS.4 Semiformal modular design  
ATE\_FUN.1 Functional testing

Developer action elements:

ATE\_DPT.3.1D The developer shall provide the analysis of the depth of testing.

Content and presentation elements:

ATE\_DPT.3.1C The analysis of the depth of testing shall demonstrate the correspondence between the tests in the test documentation and the TSF subsystems and modules in the TOE design.

ATE\_DPT.3.2C The analysis of the depth of testing shall demonstrate that all TSF subsystems in the TOE design have been tested.

ATE\_DPT.3.3C The analysis of the depth of testing shall demonstrate that all TSF modules in the TOE design have been tested.

Evaluator action elements:

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

### **5.3.4.3 ATE\_FUN.1 Functional testing**

Dependencies: ATE\_COV.1 Evidence of coverage

Developer action elements:

ATE\_FUN.1.1D The developer shall test the TSF and document the results.

ATE\_FUN.1.2D The developer shall provide test documentation.

Content and presentation elements:

ATE\_FUN.1.1C The test documentation shall consist of test plans, expected test results and actual test results.

ATE\_FUN.1.2C The test plans shall identify the tests to be performed and describe the scenarios for performing each test. These scenarios shall include any ordering dependencies on the results of other tests.

ATE\_FUN.1.3C The expected test results shall show the anticipated outputs from a successful execution of the tests.

ATE\_FUN.1.4C The actual test results shall be consistent with the expected test results.

Evaluator action elements:

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

### **5.3.4.4 ATE\_IND.2 Independent testing - sample**

Dependencies: ADV\_FSP.2 Security-enforcing functional specification  
AGD\_OPE.1 Operational user guidance  
AGD\_PRE.1 Preparative procedures  
ATE\_COV.1 Evidence of coverage  
ATE\_FUN.1 Functional testing

Developer action elements:

ATE\_IND.2.1D The developer shall provide the TOE for testing.

Content and presentation elements:

ATE\_IND.2.1C The TOE shall be suitable for testing.

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

Evaluator action elements:

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

ATE\_IND.2.2E The evaluator shall execute a sample of tests in the test documentation to verify the developer test results.

ATE\_IND.2.3E The evaluator shall test a subset of the TSF to confirm that the TSF operates as specified.

### **5.3.5 Class AVA: Vulnerability assessment**

#### **5.3.5.1 AVA\_CCA\_(EXT).1 Systematic Cryptographic Module covert channel analysis**

Dependencies: ADV\_FSP.4 Complete Functional Specification  
ADV\_IMP.1 Implementation of the TSF  
AGD\_OPE.1 Operational user guidance  
AGD\_PRE.1 Preparative User guidance

Application notes: The covert channel analysis is performed only upon the cryptographic module; a search is made for the leakage of critical cryptographic security parameters from the cryptographic module, rather than a violation of an information control policy. Inappropriate handling / leakage of any critical cryptographic security parameters (covered or not) that by design and implementation lie outside the cryptographic module is not addressed by this CCA. Thus, leakage of such parameters in such designs and implementations must be investigated by other means.

Developer action elements:

AVA\_CCA\_(EXT).1.1D For the cryptographic module, the developer shall conduct a search for covert channels for the leakage of critical cryptographic security parameters whose disclosure would compromise the security provided by the module.

Application Note: The remainder of the TOE need not be subjected to a covert channel analysis. (Ideally, a covert channel analysis on the entire TSF would determine if TSF interfaces can be used covertly for the leakage of critical cryptographic security parameters. While such extensive covert channel analysis is more complete, it is also difficult and expensive. At this time it is considered beyond the scope of effort and cost considered reasonable for COTS medium robustness products. Consequently, covert channel analysis has been limited here to the cryptographic module, but that analysis limitation does come with some added risk of unknown leakage from other parts of the TOE.

AVA\_CCA\_(EXT).1.2D The developer shall provide covert channel analysis documentation.

Content and presentation of evidence elements:

AVA\_CCA\_(EXT).1.1C The analysis documentation shall identify covert channels in the cryptographic module and estimate their capacity.

AVA\_CCA\_(EXT).1.2C The analysis documentation shall describe the procedures used for determining the existence of covert channels in the cryptographic module, and the information needed to carry out the covert channel analysis.

AVA\_CCA\_(EXT).1.3C The analysis documentation shall describe all assumptions made during the covert channel analysis.

AVA\_CCA\_(EXT).1.4C The analysis documentation shall describe the method used for estimating channel capacity, based on worst-case scenarios.

AVA\_CCA\_(EXT).1.5C The analysis documentation shall describe the worst case exploitation scenario for each identified covert channel.

AVA\_CCA\_(EXT).1.6C The analysis documentation shall provide evidence that the method used to identify covert channels is systematic.

Evaluator action elements:

AVA\_CCA\_(EXT).1.1E The NSA evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA\_CCA\_(EXT).1.2E The NSA evaluator shall confirm that the results of the covert channel analysis show that the cryptographic module meets its functional requirements.

AVA\_CCA\_(EXT).1.3E The NSA evaluator shall selectively validate the covert channel analysis through independent analysis and testing.

Application Note: The cryptographic security parameters are to be defined in the Security Target

### **5.3.5.2 AVA\_VAN.4 Methodical vulnerability analysis**

Dependencies: ADV\_ARC.1 Security architecture description  
ADV\_FSP.2 Security-enforcing functional specification  
ADV\_TDS.3 Basic modular design  
ADV\_IMP.1 Implementation representation of the TSF  
AGD\_OPE.1 Operational user guidance  
AGD\_PRE.1 Preparative procedures

Developer action elements:

AVA\_VAN.4.1D The developer shall provide the TOE for testing.

Content and presentation elements:

AVA\_VAN.4.1C The TOE shall be suitable for testing.

Evaluator action elements:

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

AVA\_VAN.4.2E The evaluator shall perform a search of public domain sources to identify potential vulnerabilities in the TOE.

AVA\_VAN.4.3E The evaluator shall perform an independent, methodical vulnerability analysis of the TOE using the guidance documentation, functional specification, TOE design, security architecture description and implementation representation to identify potential vulnerabilities in the TOE.

AVA\_VAN.4.4E The evaluator shall conduct penetration testing based on the identified potential vulnerabilities to determine that the TOE is resistant to attacks performed by an attacker possessing Moderate attack potential.

## 6 RATIONALE

---

152 This section provides the rationale for the selection of the IT security requirements, objectives, assumptions, and threats. In particular, it shows that the IT security requirements are suitable to meet the security objectives, which in turn are shown to be suitable to cover all aspects of the TOE security environment.

### 6.1 Rationale for TOE Security Objectives

**Table 6 Rationale for TOE Security Objectives**

Threat/Policy	Objectives Addressing the Threat/Policy	Rationale
<p>T.ADMIN_ERROR</p> <p>An administrator may incorrectly install or configure the TOE, or install a corrupted TOE resulting in ineffective security mechanisms.</p>	<p>O.ROBUST_ADMIN_GUIDANCE</p> <p>The TOE will provide administrators with the necessary information for secure delivery and management.</p>	<p>O.ROBUST_ADMIN_GUIDANCE (ALC_DEL.1, AGD_PRE.1, AGD_OPE.1, AGD_USR.1,) help to mitigate this threat by ensuring the TOE administrators have guidance that instructs them how to administer the TOE in a secure manner and to provide the administrator with instructions to ensure the TOE was not corrupted during the delivery process. 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.</p>

Threat/Policy	Objectives Addressing the Threat/Policy	Rationale
	<p>O.ADMIN_ROLE</p> <p>The TOE will provide administrator roles to isolate administrative actions, and to make the administrative functions available locally and remotely.</p>	<p>O.ADMIN_ROLE (FMT_SMR.2) plays a role in mitigating this threat by limiting the functions an administrator can perform in a given role.</p>
	<p>O.MANAGE</p> <p>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.</p>	<p>O.MANAGE (FMT_MTD.1(1), FMT_MTD.1(2), FMT_MTD.1(3), FMT_MTD.1(4)) also contributes to mitigating this threat by providing administrators the capability to view configuration settings. For example, if the Security Administrator made a mistake when configuring the rule-set, providing them the capability to view the rules affords them the ability to review the rules and discover any mistakes that might have been made.</p>
<p>T.ADMIN_ROGUE</p> <p>An administrator's intentions may become malicious resulting in user or TSF data being compromised.</p>	<p>O.ADMIN_ROLE</p> <p>The TOE will provide administrator roles to isolate administrative actions, and to make the administrative functions available locally and remotely.</p>	<p>O.ADMIN_ROLE (FMT_SMR.2) mitigates this threat by restricting the functions available to an administrator. This is somewhat different than the part this objective plays in countering T.ADMIN_ERROR, in that this presumes that separate individuals will be assigned separate roles.</p>

<b>Threat/Policy</b>	<b>Objectives Addressing the Threat/Policy</b>	<b>Rationale</b>
<p>T.AUDIT_COMPROMISE</p> <p>A malicious 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.</p>	<p>O.AUDIT_PROTECTION</p> <p>The TOE will provide the capability to protect audit information.</p>	<p>O.AUDIT_PROTECTION (FAU.SAR.2, FAU_STG.1-NIAP-0429, FAU_STG.3, FAU_STG.NIAP-0414-1, FMT_SMF.1) contributes to mitigating this threat by controlling access to the audit trail. The auditor and any trusted IT entities performing IDS-like functions are the only ones allowed to read the audit trail. No one is allowed to modify audit records, and the Auditor is the only one allowed to delete audit records in the audit trail. The TOE has the capability to prevent auditable actions from occurring if the audit trail is full, and of notifying an administrator if the audit trail is approaching its capacity. In addition, the TOE has the capability to restore audit data corrupted by the attacker.</p>

Threat/Policy	Objectives Addressing the Threat/Policy	Rationale
	<p>O.RESIDUAL_INFORMATION</p> <p>The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated.</p>	<p>O.RESIDUAL_INFORMATION (FDP.RIP.2) 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.</p>
	<p>O.SELF_PROTECTION</p> <p>The TSF will maintain a domain for its own execution that protects itself and its resources from external interference, tampering or unauthorized disclosure.</p>	<p>O.SELF_PROTECTION (ADV_ARC.1) The architecture will contribute 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. Likewise, ensuring that the functions that protect the audit trail are always invoked is also critical to the mitigation of this threat.</p>

Threat/Policy	Objectives Addressing the Threat/Policy	Rationale
<p><b>T.CRYPTO_COMPROMISE</b></p> <p>A malicious 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.</p>	<p><b>O.RESIDUAL_INFORMATION</b></p> <p>The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated.</p>	<p><b>O.RESIDUAL_INFORMATION (FDP_RIP.2)</b> is necessary to mitigate this threat by ensuring no TSF data remain in resources allocated to a user. 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.</p>
	<p><b>O.SELF_PROTECTION</b></p> <p>The TSF will maintain a domain for its own execution that protects itself and its resources from external interference, tampering, or unauthorized disclosure.</p>	<p><b>O.SELF_PROTECTION (ADV_ARC.1)</b> The architecture will contribute 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 cryptographic data and executable code.</p>

Threat/Policy	Objectives Addressing the Threat/Policy	Rationale
	<p>O.DOCUMENT_KEY_LEAKAGE</p> <p>The bandwidth of channels that can be used to compromise key material shall be documented.</p>	<p>O.DOCUMENT_KEY_LEAKAGE (AVA_CCA_(EXT).1) addresses this threat by requiring the developer to perform an analysis that documents the amount of key information that can be leaked via a covert channel. This provides information that identifies how much material could be inappropriately obtained within a specified time period.</p>
<p>T.FLAWED_DESIGN</p>	<p>O.CHANGE_MANAGEMENT</p> <p>The configuration of, and all changes to, the TOE and its development evidence will be analyzed, tracked, and controlled throughout the TOE's development.</p>	<p>O.CHANGE_MANAGEMENT (ALC_CMC.4, ALC_CMS.4, ALC_DVS.1, ALC_FLR.2, ALC_LCD.1) plays a role in countering this threat by requiring the developer to provide control of the changes made to the TOE's design. This includes controlling physical access to the TOE's development area, and having an automated configuration management system that ensures changes made to the TOE go through an approval process and only those persons that are authorized can make changes to the TOE's design and its documentation.</p>

Threat/Policy	Objectives Addressing the Threat/Policy	Rationale
	<p><b>O.SOUND_DESIGN</b></p> <p>The TOE will be designed using sound design principles and techniques. The TOE design, design principles and design techniques will be adequately and accurately documented.</p>	<p><b>O.SOUND_DESIGN</b> (ADV_ARC.1, ADV_FSP.5, ADV_TDS.4, ADV_INT.3,) counters this threat, to a degree, by requiring that the TOE be developed using sound engineering principles. By accurately and completely documenting the design of the security mechanisms in the TOE, including a security model, the design of the TOE can be better understood, which increases the chances that design errors will be discovered.</p>
	<p><b>O.VULNERABILITY_ANALYSIS_TEST</b></p> <p>The TOE will undergo appropriate independent vulnerability analysis and penetration testing to demonstrate the design and implementation of the TOE does not allow attackers with medium attack potential to violate the TOE's security policies.</p>	<p><b>O.VULNERABILITY_ANALYSIS_TEST</b> (AVA_VAN.4) ensures that the design of the TOE is independently analyzed for design flaws. Having an independent party perform the assessment ensures an objective approach is taken and may find errors in the design that would be left undiscovered by developers that have a preconceived incorrect understanding of the TOE's design.</p>

<b>Threat/Policy</b>	<b>Objectives Addressing the Threat/Policy</b>	<b>Rationale</b>
<p>T.FLAWED_IMPLEMENTATION</p> <p>Unintentional or intentional errors in requirements specification or design of the TOE may occur, leading to flaws that may be exploited by a malicious user or program.</p>	<p>O.CHANGE_MANAGEMENT</p> <p>The configuration of, and all changes to, the TOE and its development evidence will be analyzed, tracked, and controlled throughout the TOE's development.</p>	<p>O.CHANGE_MANAGEMENT (ALC_CMC.4, ALC_CMS.4, ALC_DVS.1, ALC_FLR.2, ALC_LCD.1)</p> <p>This objective plays a role in mitigating this threat in the same way that the flawed design threat is mitigated. By controlling who has access to the TOE's implementation representation and ensuring that changes to the implementation are analyzed and made in a controlled manner, the threat of intentional or unintentional errors being introduced into the implementation are reduced.</p>

Threat/Policy	Objectives Addressing the Threat/Policy	Rationale
	<p><b>O.SOUND_IMPLEMENTATION</b></p> <p>The implementation of the TOE will be an accurate instantiation of its design, and is adequately and accurately documented.</p>	<p>In addition to documenting the design so that implementers have a thorough understanding of the design, O.SOUND_IMPLEMENTATION (ADV_IMP.1, ADV_TDS.4, ADV_INT.3, ADV_ARC.1, ALC_TAT.1) requires that the developer's tools and techniques for implementing the design are documented. Having accurate and complete documentation, and having the appropriate tools and procedures in the development process helps reduce the likelihood of unintentional errors being introduced into the implementation.</p>
	<p><b>O.THOROUGH_FUNCTIONAL_TESTING</b></p> <p>The TOE will undergo appropriate security functional testing that demonstrates the TSF satisfies the security functional requirements.</p>	<p>Although the previous three objectives help minimize the introduction of errors into the implementation, O.THOROUGH_FUNCTIONAL_TESTING (ATE_COV.2, ATE_FUN.1, ATE_DPT.3, ATE_IND.2) 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.</p>

Threat/Policy	Objectives Addressing the Threat/Policy	Rationale
	<p>O.VULNERABILITY_ANALYSIS_TEST</p> <p>The TOE will undergo appropriate independent vulnerability analysis and penetration testing to demonstrate the design and implementation of the TOE does not allow attackers with medium attack potential to violate the TOE's security policies.</p>	<p>O.VULNERABILITY_ANALYSIS_TEST (AVA_VAN.4) 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. Having an independent party perform a vulnerability analysis and conduct testing outside the scope of functional testing increases the likelihood of finding errors.</p>
<p>T.MALICIOUS_TSF_COMPROMISE</p> <p>A malicious user or process may cause TSF data or executable code to be inappropriately accessed (viewed, modified, or deleted).</p>	<p>O.RESIDUAL_INFORMATION</p> <p>The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated.</p>	<p>O.RESIDUAL_INFORMATION (FDP_RIP.2, FCS_CKM.4) is necessary to mitigate this threat by ensuring no TSF data remain in resources allocated to a user. 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.</p>

<b>Threat/Policy</b>	<b>Objectives Addressing the Threat/Policy</b>	<b>Rationale</b>
	<p>O.SELF_PROTECTION</p> <p>The TSF will maintain a domain for its own execution that protects itself and its resources from external interference, tampering or unauthorized disclosure.</p>	<p>O.SELF_PROTECTION (ADV_ARC.1) requires that the TSF be able to protect itself from tampering and that the security mechanisms in the TSF cannot be bypassed. Without this objective, there could be no assurance that users could not view or modify TSF data or TSF executables.</p>

Threat/Policy	Objectives Addressing the Threat/Policy	Rationale
	<p>O.MANAGE</p> <p>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.</p>	<p>O.MANAGE</p> <p>(FMT_MTD.1(1)FMT_MTD.1(2), FMT_MTD.1(3), FMT_MTD.1(4), FMT_MSA.1(1), FMT_MSA.1(2), FMT_MSA.3(1), FMT_MSA.3(2), FMT_MOF.1(1), FMT_MOF.1(2), FMT_MOF.1(3), FMT_MOF.1(4), FMT_MOF.1(5), FMT_MOF.1(6), FMT_MOF.1(7), FMT_MTD.2(1), FMT_MTD.2(2), FMT_MTD.2(3), FMT_SMF.1) provides the capability to restrict access to TSF to those that are authorized to use the functions. Satisfaction of this objective (and its associated requirements) prevents unauthorized access to TSF functions and data through the administrative mechanisms.</p>

Threat/Policy	Objectives Addressing the Threat/Policy	Rationale
	<p>O.DISPLAY_BANNER</p> <p>The TOE will display an advisory warning regarding use of the TOE.</p>	<p>O.DISPLAY_BANNER (FTA_TAB.1) helps mitigate this threat by providing the Administrator the ability to remove product information (e.g., product name, version number) from a banner that is displayed to users. Having product information about the TOE provides an attacker with information that may increase their ability to compromise the TOE.</p>

Threat/Policy	Objectives Addressing the Threat/Policy	Rationale
	<p>O.TRUSTED_PATH</p> <p>The TOE will provide a means to ensure that users are not communicating with some other entity pretending to be the TOE when supplying identification and authentication data.</p>	<p>O.TRUSTED_PATH (FTP_TRP.1(1), FTP_TRP.1(2), FTP_ITC.1(1), FTP_ITC.1(2)) plays a role in addressing this threat by ensuring that there is a trusted communication path between the TSF and various users (remote administrators, and trusted IT entities (for performing replication, for instance)). This ensures the transmitted data cannot be compromised or disclosed during the duration of the trusted path. The protection offered by this objective is limited to TSF data, including authentication data and all data sent or received by trusted IT entities (a relying party's user data is not protected; only the authentication portion of the session is protected).</p>

<b>Threat/Policy</b>	<b>Objectives Addressing the Threat/Policy</b>	<b>Rationale</b>
<p><b>T.MASQUERADE</b></p> <p>A malicious user, process, or external IT entity may masquerade as an authorized entity in order to gain access to data or TOE resources.</p>	<p><b>O.ROBUST_TOE_ACCESS</b></p> <p>The TOE will provide mechanisms that control a user’s logical access to the TOE and to explicitly deny access to specific users when appropriate.</p>	<p><b>O.ROBUST_TOE_ACCESS</b> (FIA_AFL.1, FIA_ATD.1(1), FIA_ATD.1(2), FIA_UID.2, FIA_UAU.2, FIA_UAU_(EXT).5, FTA_TSE.1, AVA_SOF.1) 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 mechanisms, 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.</p>

<b>Threat/Policy</b>	<b>Objectives Addressing the Threat/Policy</b>	<b>Rationale</b>
<p>T.POOR_TEST</p> <p>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.</p>	<p>O.CORRECT_TSF_OPERATION</p> <p>The TOE will provide a capability to test the TSF to ensure the correct operation of the TSF in its operational environment.</p>	<p>While these testing activities are necessary 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.</p> <p>O.CORRECT_TSF_OPERATION (FPT_TST_(EXT).1, FPT_TST.1(1) and FPT_TST1(2)) ensures that once the TOE is installed at a customer's location, the capability exists that the integrity of the TSF (hardware and software, including the cryptographic functions) can be demonstrated, and thus providing end users the confidence that the TOE's security policies continue to be enforced.</p>

Threat/Policy	Objectives Addressing the Threat/Policy	Rationale
	<p>O.THOROUGH_FUNCTIONAL_TESTING</p> <p>The TOE will undergo appropriate security functional testing that demonstrates the TSF satisfies the security functional requirements.</p>	<p>Design analysis determines that 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.</p> <p>O.THOROUGH_FUNCTIONAL_TESTING (ATE_FUN.1, ATE_COV.2, ATE_DPT.3, ATE_IND.2) ensures that adequate functional testing is performed to demonstrate the TSF satisfies the security functional requirements and that the TOE's security mechanisms operate as documented. While functional testing serves an important purpose, it does not ensure the TSFI cannot be used in unintended ways to circumvent the TOE's security policies.</p>

<b>Threat/Policy</b>	<b>Objectives Addressing the Threat/Policy</b>	<b>Rationale</b>
	<p>O.VULNERABILITY_ANALYSIS_TEST</p> <p>The TOE will undergo appropriate independent vulnerability analysis and penetration testing to demonstrate the design and implementation of the TOE does not allow attackers with medium attack potential to violate the TOE's security policies.</p>	<p>O.VULNERABILITY_ANALYSIS_TEST (AVA_VAN.4) 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.</p>
<p>T.REPLAY</p> <p>A user may gain inappropriate access to the TOE by replaying authentication information, or may cause the TOE to be inappropriately configured by replaying TSF data or security attributes (e.g., captured as transmitted during the course of legitimate use).</p>	<p>O.REPLAY_DETECTION</p> <p>The TOE will provide a means to detect and reject the replay of authentication data and other TSF data and security attributes.</p>	<p>O.REPLAY_DETECTION (FPT_RPL.1) prevents a user from replaying authentication data. Prevention of replay of authentication data will counter the threat that a user will be able to record an authentication session between a trusted entity (administrative user or trusted IT entity) and then replay it to gain access to the TOE, and counter the ability of a user to act as another user.</p>

<b>Threat/Policy</b>	<b>Objectives Addressing the Threat/Policy</b>	<b>Rationale</b>
<p>T.RESIDUAL_DATA</p> <p>A user or process may gain unauthorized access to data through reallocation of TOE resources from one user or process to another.</p>	<p>O.RESIDUAL_INFORMATION</p> <p>The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated.</p>	<p>O.RESIDUAL_INFORMATION (FDP_RIP.2, FCS_CKM.4) 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. This means that network packets sent in response to a request will not have residual data from another packet (potentially from another user) due to the padding of a packet. The TSF data will be zeroized once the resources are released by a user/process.</p>

<b>Threat/Policy</b>	<b>Objectives Addressing the Threat/Policy</b>	<b>Rationale</b>
<p>T.RESOURCE_EXHAUSTION</p> <p>A malicious process or user may block others from system resources (e.g., connection state tables, TCP connections) via a resource exhaustion denial of service attack.</p>	<p>O.RESOURCE_SHARING</p> <p>The TOE shall provide mechanisms that mitigate attempts to exhaust connection-oriented resources provided by the TOE (e.g., entries in a connection state table; Transmission Control Protocol (TCP) connections to the TOE).</p>	<p>O.RESOURCE_SHARING (FRU_RSA.1(1), FRU_RSA.1(2), FMT_MTD.2(1), FMT_MTD.2(2) FMT_MTD.2(3)) mitigates this threat by requiring the TOE to provide controls relating to three different resources: CPU time, audit storage and available network connections. The administrator is allowed to specify a percentage of audit usage prior to needed action. The administrator is allowed to specify a percentage of processor time that is allowed to be used so that an attempt to exhaust the resource will fail when it reaches the quota. This objective also addresses the denial-of-service attack of a user attempting to exhaust the connection-oriented resources by generating a large number of half-open connections (e.g., SYN attack).</p>

Threat/Policy	Objectives Addressing the Threat/Policy	Rationale
<p><b>T.SPOOFING</b></p> <p>A malicious user, process, or external IT entity may misrepresent itself as the TOE to obtain identification and authentication data.</p>	<p><b>O.TRUSTED_PATH</b></p> <p>The TOE will provide a means to ensure that users are not communicating with some other entity pretending to be the TOE when supplying identification and authentication data.</p>	<p>It is possible for an entity other than the TOE (a subject on the TOE, or another IT entity on the network between the TOE and the end user) to provide an environment that may lead a user to mistakenly believe they are interacting with the TOE, thereby fooling the user into divulging identification and authentication information.</p> <p><b>O.TRUSTED_PATH</b> (FTP_ITC.1(1), FTP_ITC.1(2), FTP_TRP.1(1), FTP_TRP.1(2)) mitigates this threat by ensuring users have the capability to ensure they are communicating with the TOE when providing identification and authentication data to the TOE.</p>

<b>Threat/Policy</b>	<b>Objectives Addressing the Threat/Policy</b>	<b>Rationale</b>
<p><b>T.EAVESDROP</b></p> <p>A malicious user or process may observe or modify user or TSF data transmitted between physically separated parts of the TOE.</p>	<p><b>O.CRYPTOGRAPHIC_FUNCTIONS</b></p> <p>The TOE shall provide cryptographic functions (i.e., encryption/decryption and digital signature operations) to maintain the confidentiality and allow for detection of modification of TSF data that is transmitted between physically separated portions of the TOE, or stored outside the TOE.</p>	<p><b>O.CRYPTOGRAPHIC_FUNCTIONS</b> (FCS_CKM.1(1), FCS_CKM.1(2), FCS_CKM.2, FCS_CKM.4, FCS_COP.1(1), FCS_COP.1(2) mitigates this threat by providing for the use of cryptographic functions to detect when information has been modified.</p>
	<p><b>O.PROTECT_IN_TRANSIT</b></p> <p>The TSF shall protect TSF data when it is in transit between the TSF and another trusted IT entity.</p>	<p><b>O.PROTECT_IN_TRANSIT</b> (FPT_ITA.1, FPT_ITC.1, FPT_ITL.1, FTP_TRP.1(1), FTP_TRP.1(2), FTP_ITC.1(1), FTP_ITC.1(2) satisfies this threat by ensuring protection of the communication between the TOE and trusted IT entities while transmitting data.)</p>

Threat/Policy	Objectives Addressing the Threat/Policy	Rationale
<p>T.UNATTENDED_SESSION</p> <p>A user may gain unauthorized access to an unattended session.</p>	<p>O.ROBUST_TOE_ACCESS</p> <p>The TOE will provide mechanisms that control a user's logical access to the TOE and to explicitly deny access to specific users when appropriate.</p>	<p>O. ROBUST_TOE_ACCESS (FTA_SSL.3) helps to mitigate this threat by including mechanisms that place controls on user's sessions. Local and remote administrator's sessions are dropped after an Administrator-defined time period of inactivity. Dropping the connection of a local and remote session (after the specified time period) reduces the risk of someone accessing the local and remote machines where the session was established, thus gaining unauthorized access to the session.</p>

<b>Threat/Policy</b>	<b>Objectives Addressing the Threat/Policy</b>	<b>Rationale</b>
<p>T.UNAUTHORIZED_ACCESS</p> <p>A user may gain access to user data for which they are not authorized according to the TOE security policy.</p>	<p>O.MEDIATE_INFORMATION_FLOW</p> <p>The TOE must mediate the flow of information between sets of TOE network interfaces or between a network interface and the TOE itself in accordance with its security policy.</p>	<p>O.MEDIATE_INFORMATION_FLOW (FDP_IFC.1(1), FDP_IFC.1(2), FDP_IFF.1(1), FDP_IFF.1(2), works to mitigate this threat by ensuring that all network packets that flow through the TOE are subject to the information flow policies. One of the rules ensures that the network identifiers in a packet is in the set of network identifiers associated with a TOE's network interface. Therefore, if a user supplied a network identifier in a packet that purported to originate from a network associated with a TOE network interface other than the one the user supplied the packet on, the packet would not be allowed to flow through the TOE or access TOE services. The authenticated TOE policy ensures that user data being sent between PEER TOEs is encrypted if there is a rule (specified by the Security Administrator) that states</p>

<b>Threat/Policy</b>	<b>Objectives Addressing the Threat/Policy</b>	<b>Rationale</b>
		<p>data is to be encrypted between those two hosts.</p> <p>The authenticated TOE policy allows the administrator to specify each originating host (identified by IP address), which destination addresses must be access through a router and which destination addresses may be accessed without encryption. If a potential security violation has been detected, the TOE displays a message that identifies the potential security violation to all administrator consoles. The consoles include the local TOE console and any active remote administrative sessions. If an administrator is not currently accessing the TOE, the message is stored and immediately displayed the next time an administrator accesses the TOE.</p> <p>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</p>

Threat/Policy	Objectives Addressing the Threat/Policy	Rationale
		ensures that no other user can modify the information flow policy to bypass the intended TOE security policy.
	<p>O.USER_GUIDANCE</p> <p>The TOE will provide users with the information necessary to correctly use the security mechanisms.</p>	<p>O.USER_GUIDANCE (AGD_USR.1) mitigates this threat by providing the user the information necessary to use the security mechanisms that control access to user data in a secure manner. For instance, the method by which the discretionary access control mechanism (FDP_ACC.1, FDP_ACF.1) is configured, and how to apply it to the data the user owns, is described in the user guidance. If this information were not available to the user, the information may be left unprotected, or the user may mis-configure the controls and unintentionally allow unauthorized access to their data.</p>

<b>Threat/Policy</b>	<b>Objectives Addressing the Threat/Policy</b>	<b>Rationale</b>
<p><b>T.UNIDENTIFIED_ACTIONS</b></p> <p>The administrator may fail to notice potential security violations, thus limiting the administrator's ability to identify and take action against a possible security breach.</p>	<p><b>O.AUDIT_REVIEW</b></p> <p>The TOE will provide the capability to selectively view audit information, and alert the administrator of identified potential security violations.</p>	<p><b>O.AUDIT_REVIEW</b> (FAU_SAA.1-NIAP-0407, FAU_ARP.1, FAU_ARP_ACK_(EXT).1, FAU_SAR.1, FAU_SAR.3) helps to mitigate this threat by providing a variety of mechanisms for monitoring the use of the system. The two basic ways audit review is performed is through analysis of the audit trail produced by the audit mechanism, and through the use of an automated analysis and alarm system.</p> <p>For analyzing the audit trail, the TOE requires an Auditor role. This role is restricted to Audit record review and the deletion of the audit trail for maintenance purposes. A search and sort capability provides an efficient mechanism for the Audit Administrator to view pertinent audit information.</p> <p>In addition to the local Auditor role, the TOE also</p>

<b>Threat/Policy</b>	<b>Objectives Addressing the Threat/Policy</b>	<b>Rationale</b>
		<p>has the capability to export the audit information to an external audit analysis tool (such as an intrusion detection system) for more detailed or composite audit analysis.</p> <p>The TOE's audit analysis mechanism must consist of a minimum set of configurable audit events that could indicate a potential security violation. Thresholds for these events must be configurable by an appropriate administrative role. By configuring these auditable events, the TOE monitors the occurrences of these events (e.g. set number of authentication failures, set number directory access failures, self-test failures, etc.) and immediately notifies an administrator once an event has occurred or a set threshold has been met.</p>

<b>Threat/Policy</b>	<b>Objectives Addressing the Threat/Policy</b>	<b>Rationale</b>
<p>T.UNAUTHORIZED_PEER</p> <p>An unauthorized IT entity may attempt to establish a security association with the TOE.</p>	<p>O.PEER_AUTHENTICATION</p> <p>The TOE will authenticate each peer TOE that attempts to establish a security association with the TOE.</p>	<p>O.PEER_AUTHENTICATION (FCS_IKE_(EXT).1) mitigates this threat by requiring that the TOE implement the Internet Key Exchange protocol, as specified in RFC2409, to establish a secure, authenticated channel between the TOE and another remote router before establishing a security association with that router.</p>
<p>T.UNKNOWN_STATE</p> <p>When the TOE is initially started or restarted after a failure, the security state of the TOE may be unknown.</p>	<p>O.MAINT_MODE</p> <p>The TOE shall provide a mode from which recovery or initial startup procedures can be performed.</p>	<p>O.MAINT_MODE (FPT_RCV.2) helps to mitigate this threat by ensuring that the TOE does not continue to operate in an insecure state when a hardware or software failure occurs. After a failure, the TOE enters a state that disallows operations and requires an administrator to follow documented procedures to return the TOE to a secure state.</p>

Threat/Policy	Objectives Addressing the Threat/Policy	Rationale
	<p>O.CORRECT_TSF_OPERATION</p> <p>The TOE will provide a capability to test the TSF to ensure the correct operation of the TSF in its operational environment.</p>	<p>O.CORRECT_TSF_OPERATION (FPT_TST_EXT).1, FPT_TST.1(1) and FPT_TST.1(2)) counters this threat by ensuring that the TSF runs a suite of tests to successfully demonstrate the correct operation of the TSF (hardware and software) and the TSF's cryptographic components at initial startup of the TOE. In addition to ensuring that the TOE's security state can be verified, an administrator can verify the integrity of the TSF's data and stored code and the TSF's cryptographic data and stored code using the TOE-provided cryptographic mechanisms.</p>
	<p>O.SOUND_DESIGN</p> <p>The TOE will be designed using sound design principles and techniques. The TOE design, design principles and design techniques will be adequately and accurately documented.</p>	<p>O.SOUND_DESIGN (ADV_ARC.1) works to mitigate this threat by requiring that the TOE developers provide accurate and complete design documentation of the security mechanisms in the TOE, including a security model. By providing this documentation, the possible secure states of the TOE are described, thus enabling the administrator to return the TOE to one of these states during the recovery process.</p>

Threat/Policy	Objectives Addressing the Threat/Policy	Rationale
	<p>O.ROBUST_ADMIN_GUIDANCE</p> <p>The TOE will provide administrators with the necessary information for secure delivery and management.</p>	<p>O. ROBUST_ADMIN_GUIDANCE (ADO_PRE.1, AGD_OPE.1) provides administrative guidance for the secure start-up of the TOE and guidance to configure and administer the TOE securely. This guidance provides administrators with the information necessary to ensure that the TOE is started and initialized in a secure manor. The guidance also provides information about the corrective measure necessary when a failure occurs (i.e., how to bring the TOE back into a secure state).</p>
<p>P.ACCESS_BANNER</p> <p>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 TOE.</p>	<p>O.DISPLAY_BANNER</p> <p>The TOE will display an advisory warning regarding use of the TOE.</p>	<p>O.DISPLAY_BANNER (FTA_TAB.1) satisfies this policy by ensuring that the TOE displays an Administrator-configurable banner that provides all users with a warning about the unauthorized use of the TOE. This is required to be displayed before an interactive administrative session.</p>

<b>Threat/Policy</b>	<b>Objectives Addressing the Threat/Policy</b>	<b>Rationale</b>
<p><b>P.ACCOUNTABILITY</b></p> <p>The authorized users of the TOE shall be held accountable for their actions within the TOE.</p>	<p><b>O.AUDIT_GENERATION</b></p> <p>The TOE will provide the capability to detect and create records of security-relevant events associated with users.</p>	<p><b>O.AUDIT_GENERATION</b> (FAU_GEN.1-NIAP-0429, FAU_GEN.2-NIAP-410, FIA_USB.1, FAU_SEL.1-NIAP-0407) addresses this policy by providing an audit mechanism to record the actions of a specific user, and the capability for an administrator to “pre-select” audit events based on the user ID. The audit event selection function is configurable during run-time to ensure the TOE is able to capture security-relevant events given changes in threat conditions. 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.). Attributes used in the audit record generation process are also required to be bound to the subject, ensuring users are held accountable</p>

Threat/Policy	Objectives Addressing the Threat/Policy	Rationale
	<p><b>O.TIME_STAMPS</b></p> <p>The TOE shall provide reliable time stamps and the capability for the administrator to set the time used for these time stamps.</p>	<p><b>O.TIME_STAMPS</b> (FPT_STM.1, FMT_MTD.1) plays a role in supporting this policy by requiring the TOE to provide a reliable time stamp. 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.</p>
	<p><b>O.ROBUST_TOE_ACCESS</b></p> <p>The TOE will provide mechanisms that control a user's logical access to the TOE and to explicitly deny access to specific users when appropriate.</p>	<p><b>O.ROBUST_TOE_ACCESS</b> (FIA_UID.2, FIA_UAU.2, FIA_UAU_(EXT).5) 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.</p>

Threat/Policy	Objectives Addressing the Threat/Policy	Rationale
<p><b>P.ADMIN_ACCESS</b></p> <p>Administrators shall be able to administer the TOE both locally and remotely through protected communications channels.</p>	<p><b>O.ADMIN_ROLE</b></p> <p>The TOE will provide administrator roles to isolate administrative actions, and to make the administrative functions available locally and remotely.</p>	<p><b>O.ADMIN_ROLE (FMT_SMR.2)</b> supports this policy by requiring the TOE to provide mechanisms (e.g., local authentication, remote authentication, means to configure and manage the TOE both remotely and locally) that allow remote and local administration of the TOE. This is not to say that everything that can be done by a local administrator must also be provided to the remote administrator. In fact, it may be desirable to have some functionality restricted to the local administrator.</p>
	<p><b>O.TRUSTED_PATH</b></p> <p>The TOE will provide a means to ensure that users are not communicating with some other entity pretending to be the TOE when supplying identification and authentication data.</p>	<p><b>O.TRUSTED_PATH (FTP_TRP.1(1), FTP_TRP.1(2), FTP_ITC.1(1), FTP_ITC.1(2))</b> satisfies this policy by requiring that each remote administrative and management session for all trusted users is authenticated and conducted via a secure channel. Additionally, all trusted IT entities (e.g., trusted peer directories, intrusion detection systems) connect through a protected channel, thus avoiding disclosure and spoofing problems.</p>

<b>Threat/Policy</b>	<b>Objectives Addressing the Threat/Policy</b>	<b>Rationale</b>
<p><b>P.COMPATIBILITY</b></p> <p>The TOE must meet RFC requirements for implemented protocols to facilitate inter-operation with other routers and network equipment using the same protocols.</p>	<p><b>O.PROTOCOLS</b></p> <p>The TOE will ensure that standardized protocols are implemented in the TOE to RFC and/or Industry specifications to ensure interoperability.</p>	<p><b>O.PROTOCOLS</b> (FPT_FLS.1, FPT_PRO_(EXT).1) satisfies this policy by requiring that standardized protocols are implemented in the TOE to ensure interoperability among peer TOEs therefore not compromising the secure state of the router.</p>
<p><b>P.CRYPTOGRAPHY</b></p> <p>The TOE shall use NIST FIPS validated cryptography as a baseline with additional NSA-approved methods for key management (i.e.; generation, access, distribution, destruction, handling, and storage of keys), and for cryptographic operations (i.e.; encryption, decryption, signature, hashing, key exchange, and random number generation services).</p>	<p><b>O.CRYPTOGRAPHIC_FUNCTIONS</b></p> <p>The TOE shall use NIST FIPS validated cryptography as a baseline with additional NSA-approved methods for key management (i.e.; generation, access, distribution, destruction, handling, and storage of keys), and for cryptographic operations (i.e.; encryption, decryption, signature, hashing, key exchange, and random number generation services).</p>	<p><b>O.CRYPTOGRAPHIC_FUNCTIONS</b> (FCS_CKM.1(1), FCS_CKM.1(2), FCS_CKM.2, FCS_CKM.4, FCS_CKM_(EXT).2, FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3), FCS_COP.1(4), FCS_COP.1(5)) FCS_COP_(EXT).1 implements this policy, requiring a combination of FIPS-validation and non-FIPS-validated cryptographic mechanisms that are used to provide encryption/decryption services, and digital signature functions. Functions include symmetric encryption and decryption, digital signatures, and key generation and establishment functions.</p>

Threat/Policy	Objectives Addressing the Threat/Policy	Rationale
	<p>O.RESIDUAL_INFORMATION</p> <p>The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated or upon completion of a function that residual biometric data could not be reused.</p>	<p>O.RESIDUAL_INFORMATION (FDP_RIP.2, FCS_CKM.4) 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.</p>
<p>P.VULNERABILITY_ANALYSIS_TEST</p> <p>The TOE must undergo appropriate independent vulnerability analysis and penetration testing to demonstrate that the TOE is resistant to an attacker possessing a medium attack potential.</p>	<p>O.VULNERABILITY_ANALYSIS_TEST</p> <p>The TOE will undergo appropriate independent vulnerability analysis and penetration testing to demonstrate the design and implementation of the TOE does not allow attackers with medium attack potential to violate the TOE's security policies.</p>	<p>O.VULNERABILITY_ANALYSIS_TEST (AVA_VAN.4) satisfies this policy by ensuring that an independent analysis is performed on the TOE and penetration testing based on that analysis is performed. Having an independent party perform the analysis helps ensure objectivity and eliminates preconceived notions of the TOE's design and implementation that may otherwise affect the thoroughness of the analysis. The level of analysis and testing requires that an attacker with a moderate attack potential cannot compromise the TOE's ability to enforce its security policies.</p>

## 6.2 Rationale for the Security Objectives for the Environment

This PP does not require any security assurance requirements on the below non-IT environment objectives. The assumption and objectives all rely on assurance measures such as policies and procedures, which are outside the scope of this PP.

**Table 7 Rational for IT Environmental Objectives**

Assumption	Environmental Objective Addressing the Assumption	Rationale
<p><b>A.AVAILABILITY</b> Network resources shall be available to allow clients to satisfy mission requirements and to transmit information.</p>	<p><b>OE.AVAILABILITY</b> Network resources will be available to allow clients to satisfy mission requirements and to transmit information.</p>	<p>Network resources shall be available to allow clients to satisfy mission requirements and to transmit information.</p>
<p><b>A.NO_GENERAL_PURPOSE</b> The administrator ensures there are no general-purpose computing or storage repository capabilities (e.g., compilers, editors, or user applications) available on the TOE.</p>	<p><b>OE.NO_GENERAL_PURPOSE</b> The Administrator ensures there are no general-purpose computing or storage repository capabilities (e.g., compilers, editors, or user applications) available on the TOE.</p>	<p>The Router must not include any general-purpose computing or storage capabilities. This will protect the TSF data from malicious processes.</p>

Assumption	Environmental Objective Addressing the Assumption	Rationale
<p><b>A.PHYSICAL</b></p> <p>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.</p>	<p><b>OE.PHYSICAL</b></p> <p>Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the IT environment.</p>	<p>The TOE, the TSF data, and protected user data is assumed to be protected from physical attack (e.g., theft, modification, destruction, or eavesdropping). Physical attack could include unauthorized intruders into the TOE environment, but it does not include physical destructive actions that might be taken by an individual that is authorized to access the TOE environment.</p>

### 6.3 Rationale for TOE Security Requirements

**Table 8 Rationale for TOE Security Requirements**

Objective	Requirements Addressing the Objective	Rationale
<p>O.ADMIN_ROLE</p> <p>The TOE will provide administrator roles to isolate administrative actions, and to make the administrative functions available locally and remotely.</p>	<p>FMT_SMR.2</p>	<p>FMT_SMR.2 requires that three roles exist for administrative actions: the Security Administrator, who is responsible for configuring most security-relevant parameters on the TOE; the Cryptographic Administrator, who is responsible for managing the security data that is critical to the cryptographic operations; and the Audit Administrator, who is responsible for reading and deleting the audit trail. The TSF is able to associate a human user with one or more roles and these roles isolate administrative functions in that the functions of these roles do not overlap. It is true that the design of some systems could enable a rogue security administrator to manipulate cryptographic data by, for instance, writing directly to kernel memory. While this scenario is a security concern, this objective does not counter that aspect of T.ADMIN_ROGUE. If a security administrator were to perform such an action, the auditing requirements (along with the audit trail protection requirements) afford some measure of detectability of the rogue administrator's actions.</p>

Objective	Requirements Addressing the Objective	Rationale
<p>O.AUDIT_GENERATION</p> <p>The TOE will provide the capability to detect and create records of security-relevant events associated with users.</p>	<p>FAU_GEN.1-NIAP-0410</p>	<p>FAU_GEN.1-NIAP-0429 defines the set of events that the TOE must be capable of recording. This requirement ensures that an 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. There is a minimum of information that must be present in every audit record and this requirement defines that, and the additional information that must be recorded 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.</p>
	<p>FAU_GEN.2-NIAP-0410</p>	<p>FAU_GEN.2-NIAP-0410 ensures that the audit records associate a user identity with the auditable event. Although the FIA_ATD.1 requirements mandate that a “userid” be used to represent a user identity, the TOE developer is able to associate different types of user-ids with different users in order to meet this objective.</p>

Objective	Requirements Addressing the Objective	Rationale
	FIA_USB.1	FIA_USB.1 plays a role in 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 authenticated 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.
	FAU_SEL.1 – NIAP- 0407	FAU_SEL.1-NIAP-0407 allows the selected administrator(s) 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 and providing the ability to focus on the actions of an individual user. In addition, the requirement has been refined to require that the audit event selection function is configurable during run-time to ensure the TOE is able to capture security-relevant events given changes in threat conditions.
<b>O.AUDIT_PROTECTION</b> The TOE will provide the capability to protect audit information.	FMT_MOF.1(3) FAU_SAR.2 FAU_STG.1-NIAP-0429 FAU_STG.3 FAU_STG.NIAP-0414-1-NIAP-0429 FMT_SMF.1	FMT_MOF.1 restricts the ability to control the behavior of the audit and alarm mechanism to the Security Administrator. The Security Administrator is the only user that controls the behavior of the events that generate alarms and whether the alarm mechanism is enabled or

Objective	Requirements Addressing the Objective	Rationale
		<p>disabled.</p> <p>FAU_SAR.2 restricts the ability to read the audit trail to the Auditor, 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 ordinary file).</p> <p>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 Audit Administrator; or if the option of overwriting old audit records is chosen by the Administrator in FAU_STG.NIAP-0414-1-NIAP-0429, the audit data may be deleted/overwritten. Since the Audit Administrator is trusted to review the audit data, the threat being countered is that the administrator does something malicious and then attempts to conceal it by configuring the audit log to overwrite old records. Presumably the administrator would then attempt to fill up the audit log in order to overwrite the thing they just did, and the fact that they reconfigured the audit log overwrite action. The Audit Administrator would hopefully notice this activity and detect the fact that the administrator was performing illicit activities. The fact that the administrator does not directly have the ability to delete the audit records helps ensure that audit records are</p>

Objective	Requirements Addressing the Objective	Rationale
		<p>kept until the Audit Administrator deems they are no longer necessary. FAU_STG.1-NIAP-0429 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.</p> <p>FAU_STG.3 requires that the administrators be alerted when the audit trail exceeds a capacity threshold established by the Security Administrator. In addition, an audit record is cut which will trigger the analysis performed in FAU_SAA, resulting in an FAU_ARP alarm being issued. This ensures that an administrator has the opportunity to manage the audit trail before it becomes full and the avoiding the possible loss of audit data.</p> <p>FAU_STG.NIAP-0414-1-NIAP-0429 allows the Security Administrator to configure the TOE so that if the audit trail does become full, either the TOE will prevent any events from occurring (other than actions taken by the administrator) that would generate an audit record or the audit mechanism will overwrite the oldest audit records with new records.</p> <p>FMT_SMF.1 requires the TOE to provide an administrator with a facility to backup, recover and archive audit data ensuring the ability to recover corrupted audit records, and access to a complete</p>

Objective	Requirements Addressing the Objective	Rationale
		history of audit information.
<p>O.AUDIT_REVIEW</p> <p>The TOE will provide the capability to selectively view audit information, and alert the administrator of identified potential security violations.</p>	<p>FAU_ARP.1</p> <p>FAU_ARP_ACK_(EXT).1</p> <p>FAU_SAA.1-NIAP-0407</p> <p>FAU_SAR.1</p> <p>FAU_SAR.3</p>	<p>FAU_SAA.1-NIAP-0407 defines the events (or rules) that indicate a potential security violation and will generate an alarm. The triggers for these events are largely configurable by the Security Administrator. Some rules are not configurable, or configurable by the cryptographic administrator.</p> <p>FAU_ARP.1 requires that the alarm be displayed at the local administrative console and at the remote administrative console(s) when auditor and security administrative session(s) exists. For alarms at remote consoles, the alarm is sent either during an established session or upon session establishment (as long as the alarm has not been acknowledged). This is required to increase the likelihood that the alarm will be received as soon as possible. This requirement also dictates the information that must be displayed with the alarm. The potential security violation is identified in the alarm, as are the contents of the audit records of the events that accumulated and triggered the alarm. The information in the audit records is necessary; it allows the administrators to react to the potential security violation without having to search through the audit trail looking for the related events.</p>

Objective	Requirements Addressing the Objective	Rationale
		<p>FAU_ARP_ACK_(EXT).1 requires that an alarm generated by the mechanism that implements the FAU_ARP requirement be maintained until an administrator acknowledges it. This ensures that the alarm message will not be obstructed and the administrators will be alerted of a potential security violation. Additionally, this requires that the acknowledgement be transmitted to users that received the alarm, thus ensuring that that set of administrators knows that the user specified in the acknowledgement message has addressed the alarm.</p> <p>FAU_SAR.1 (both iterations) is used to provide both the auditor and an external audit analysis function the capability to read the entire audit data contained in the audit trail. This requirement also mandates the audit information be presented in a manner that is suitable for the end user (auditor or external system) to interpret the audit trail. It is expected that the audit information be presented in such a way that the end user can examine an audit record and have the appropriate information (that required by FAU_GEN.2-NIAP-410) presented together to facilitate the analysis of the audit review. Ensuring the audit data are presented in an interpretable format will enhance the ability of the</p>

Objective	Requirements Addressing the Objective	Rationale
	<p style="text-align: center;">145</p>	<p>entity performing the analysis to identify potential security violations.</p> <p>FAU_SAR.3 complements FAU_SAR.1 by providing the administrators the flexibility to specify criteria that can be used to search or sort the audit records residing in the audit trail.</p> <p>FAU_SAR.3 requires the administrators be able to establish the audit review criteria based on a userid and role so that the actions of a user can be readily identified and analyzed. Allowing the administrators to perform searches or sort the audit records based on dates and times provides the capability to facilitate the administrator's review of incidents that may have taken place at a certain time. It is important to note that the intent of sorting in this requirement is to allow the administrators the capability to organize or group the records associated with a given criteria.</p>

Objective	Requirements Addressing the Objective	Rationale
<p>O.CHANGE_MANAGEMENT</p> <p>The configuration of, and all changes to, the TOE and its development evidence will be analyzed, tracked, and controlled throughout the TOE's development.</p>	<p>ALC_CMC.4</p> <p>ALC_CMS.4</p> <p>ALC_DVS.1</p> <p>ALC_FLR.2</p> <p>ALC_LCD.1</p>	<p>ALC_CMC.4 contributes to this objective by requiring the developer have a configuration management plan that describes how changes to the TOE and its evaluation deliverables are managed. The developer is also required to employ a configuration management system that operates in accordance with the CM plan and provides the capability to control who on the development staff can make changes to the TOE and its developed evidence. This requirement also ensures that authorized changes to the TOE have been analyzed and the developer's acceptance plan describes how this analysis is performed and how decisions to incorporate the changes to the TOE are made</p> <p>ALC_CSC.4 is necessary to define what items must be under the control of the CM system. This requirement ensures that the TOE implementation representation, design documentation, test documentation (including the executable test suite), user and administrator guidance, CM documentation and security flaws are tracked by the CM system.</p>

Objective	Requirements Addressing the Objective	Rationale
		<p>ALC_DVS.1 requires the developer describe the security measures they employ to ensure the integrity and confidentiality of the TOE is maintained. The physical, procedural, and personnel security measures the developer uses provides an added level of control over who and how changes are made to the TOE and its associated evidence.</p> <p>ALC_FLR.2 plays a role in satisfying the "analyzed" portion of this objective by requiring the developer to have procedures that address flaws that have been discovered in the product, either through developer actions (e.g., developer testing) or those discovered by others. The flaw remediation process used by the developer corrects any discovered flaws and performs an analysis to ensure new flaws are not created while fixing the discovered flaws.</p> <p>ALC_LCD.1 requires the developer to document the life-cycle model used in the development and maintenance of the TOE. This life-cycle model describes the procedural aspects regarding the development of the TOE, such as design methods, code or documentation reviews, how changes to the TOE are reviewed and accepted or rejected.</p>

<b>Objective</b>	<b>Requirements Addressing the Objective</b>	<b>Rationale</b>
		<p>ALC_CMC.4 and ALC_CMS.4 requires that the CM system use an automated means to control changes made to the TOE. If automated tools are used by the developer to analyze, or track changes made to the TOE, those automated tools must be described. This aids in understanding how the CM system enforces the control over changes made to the TOE.</p>

Objective	Requirements Addressing the Objective	Rationale
<p><b>O.CORRECT_TSF_OPERATION</b></p> <p>The TOE will provide the capability to test the TSF to ensure the correct operation of the TSF in its operational environment.</p>	<p>FPT_TST_(EXT).1, FPT_TST.1(1) FPT_TST.1(2)</p>	<p>O_CORRECT_TSF_OPERATION requires two security functional requirements in the FPT class, FPT_TST. These functional requirements provide the end user with the capability to ensure the TOE's security mechanisms continue to operate correctly in the field. FPT_TST_(EXT).1 has been created to ensure end user tests exist to demonstrate the correct operation of the security mechanisms required by the TOE that are provided by the hardware and that the TOE's software and TSF data has not been corrupted. Hardware failures could render a TOE's software ineffective in enforcing its security policies and this requirement provides the end user the ability to discover any failures in the hardware security mechanisms. FPT_TST.1(1) and FPT_TST.1(2) are necessary to ensure the correctness of the TSF software and TSF data. 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.</p>
<p><b>O.CRYPTOGRAPHIC_FUNCTIONS</b></p> <p>The TOE shall provide cryptographic functions (i.e., encryption/decryption and digital signature operations) to maintain the</p>	<p>FCS_BCM_(EXT).1 FCS_CKM.1(1) FCS_CKM.1(2) FCS_CKM.2 FCS_CKM.4</p>	<p>The FCS requirements used in this PP satisfy this objective by levying requirements that ensure the cryptographic standards include the NIST FIPS publications (where possible) and NIST approved ANSI standards. The intent is to have the</p>

Objective	Requirements Addressing the Objective	Rationale
<p>confidentiality and allow for detection of modification of TSF data that is transmitted between physically separated portions of the TOE, or stored outside the TOE.</p>	<p>FCS_CKM_(EXT).2  FCS_COP.1(1)  FCS_COP.1(2)  FCS_COP.1(3)  FCS_COP.1(4)  FCS_COP.1(5)</p>	<p>satisfaction of the cryptographic standards be validated through a NIST FIPS 140 validation.</p> <p>The core functionality to be supported is encryption/decryption using a symmetric algorithm, and digital signature generation and verification using asymmetric algorithms. Since these operations involve cryptographic keys, how the keys are generated and/or otherwise obtained have to also be specified.</p> <p>FCS_BCM_(EXT).1 is an extended requirement that specifies not only that cryptographic functions that are FIPS-approved and must be validated by FIPS, but also what NIST FIPS rating level the cryptographic module must satisfy. The level specifies the degree of testing of the module. The higher the level, the more extensive the module is tested.</p> <p>FCS_CKM.1(1) is a requirement that a cryptomodule generate symmetric keys. Such keys are used by the TDEA or AES encryption/decryption functionality specified in FCS_COP.1(1).</p> <p>FCS_CKM.1(2) is a requirement that a cryptomodule generate asymmetric keys. Such keys are used for cryptographic signatures as specified in FCS_COP.1(2).</p> <p>FCS_CKM.1 requires that the TSF validate all keys generated to assure that it meet relevant standards.</p> <p>FCS_CKM_(EXT).2 requires that</p>

Objective	Requirements Addressing the Objective	Rationale
		<p>keys are handled appropriately and associated with the correct entities, and that transfer of keys is done with error detection. Storage of persistent secret and private keys must be done in a secure fashion.</p> <p>FCS_COP.1(3) requires that the TSF provide hashing services using a NIST-approved implementation of the Secure Hash Algorithm and</p> <p>FCS_COP.1(4) requires the TSF's message authentication services be compliant with either of the NIST-approved approaches, HMAC or CCM..</p> <p>Another way of obtaining key material for symmetric algorithms is through cryptographic key establishment, as specified in FCS_COP.1(5). Key establishment has two aspects: key agreement and key distribution. Key agreement occurs when two entities exchange public data yet arrive at a mutually shared key without ever passing that key between the two entities (for example, the Diffie-Hellman algorithm).</p> <p>Key distribution (FCS_CKM.2) occurs when the key is transmitted from one entity to the TOE. If the entity is electronic and a protocol is used to distribute the key, it is referred to in this PP as "Key Transport". If the key is loaded into the TOE it can be loaded electronically (e.g., from a floppy drive, smart card, or electronic keyfill device) or manually (e.g.,</p>

Objective	Requirements Addressing the Objective	Rationale
		<p>typed in). One or more of these methods must be selected.</p> <p>FCS_CKM.4 provides the functionality for ensuring key and key material is zeroized. This applies not only to key that resides in the TOE, but also to intermediate areas (physical memory, page files, memory dumps, etc.) where key may appear.</p> <p>FCS_COP.1(1) specifies that TDEA or AES be used to perform encryption and decryption operations. FCS_COP.1(2) gives three options for providing the digital signature capability; these requirements reference the appropriate standards for each digital signature option.</p>
<p><b>O.DISPLAY_BANNER</b> The TOE will display an advisory warning regarding use of the TOE.</p>	<p>FTA_TAB.1</p>	<p>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.</p>

Objective	Requirements Addressing the Objective	Rationale
<p>O.DOCUMENT_KEY_LEAKAGE</p> <p>The bandwidth of channels that can be used to compromise key material shall be documented.</p>	<p>AVA_CCA_(EXT).1</p>	<p>AVA_CCA_(EXT).1 requires that a covert channel analysis be performed on the entire TOE to determine the bandwidth of possible cryptographic key leakage. While there are no requirements to limit the bandwidth, the results of this analysis will provide useful guidance on what the specified lifetime of the cryptographic keys should be in order to reduce the damage due to a key compromise.</p>
<p>O.MAINT_MODE</p> <p>The TOE shall provide a mode from which recovery or initial startup procedures can be performed.</p>	<p>FPT_RCV.2</p>	<p>This objective is met by using the FPT_RCV.2 requirement, which ensures that the TOE does not continue to operate in an insecure state when a hardware or software failure occurs. Upon the failure of the TSF self-tests the TOE will no longer be assured of enforcing its security policies. Therefore, the TOE enters a state that operations cease and requires an administrator to follow documented procedures that instruct them on to return the TOE to a secure state. These procedures may include running diagnostics of the hardware, or utilities that may correct any integrity problems found with the TSF data or code. Solely specifying that the administrator reload and install the TOE software from scratch, while may be required in some cases, does not meet the intent of this requirement.</p>

Objective	Requirements Addressing the Objective	Rationale
<p><b>O.MANAGE</b></p> <p>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.</p>	<p>FMT_MSA.1(1)                      FMT_MSA.1(2)                      FMT_MSA.3(1)                      FMT_MSA.3(2)                      FMT_MSA.2                      FMT_MOF.1(1)                      FMT_MOF.1(2)                      FMT_MOF.1(3)                      FMT_MOF.1(4)                      FMT_MOF.1(5)                      FMT_MOF.1(6)                      FMT_MOF.1(7)                      FMT_MTD.1(1)                      FMT_MTD.1(2)                      FMT_MTD.1(3)                      FMT_MTD.1(4)                      FMT_SMF.1</p>	<p>The FMT requirements are used to satisfy this management objective, and other objectives that specify the control of functionality. The requirement's rationale for this objective focuses on the administrator's capability to perform management functions in order to control the behavior of security functions.</p> <p>FMT_MSA.1(1) and FMT_MSA.1(2) both provide the Security Administrator the capability to manipulate the security attributes of the objects in their scope of control that determine the access policy.</p> <p>FMT_MSA.2 Ensures that the TSF will accept only secure values for security attributes.</p> <p>FMT_MSA.3(1) requires that by default, the TOE does not allow an information flow, rather than allowing information flows until a rule in the ruleset disallows it.</p> <p>FMT_MOF.1(2) and FMT_MSA.3(2) are related to the services provided by FAU_UAU.1(1) and provide the Security Administrator control as to the availability of these services. FMT_MOF.1(2) provides the ability to enable or disable the TOE services to the Security Administrator.</p> <p>FMT_MSA.3(2) requires that these services by default are disabled. Since the Security Administrator must explicitly enable these services it</p>

Objective	Requirements Addressing the Objective	Rationale
		<p>ensures the Security Administrator is aware that they are running. This requirement does afford the Security Administrator the capability to override this restrictive default and allow the services to be started whenever the TOE reboots or is restarted.</p> <p>FMT_MOF.1(1) is used to ensure the administrators have the ability to invoke the TOE self-tests at any time. The ability to invoke the self-tests is provided to all administrators. The Security Administrator is able to modify the behavior of the tests (e.g., select when they run, select a subset of the tests).</p> <p>FMT_MOF.1(3) specifies the ability of the administrators to control the security functions associated with audit and alarm generation. The ability to control these functions has been assigned to the appropriate administrative roles.</p> <p>FMT_MOF.1(6) This requirement limits the ability to manipulate the values that are used in the FRU_RSA.1(2) requirements to the Security Administrator. The Security Administrator is provided the capability to assign the network identifier(s) they wish to place resource restrictions on and allows them to also specify over what period</p>

Objective	Requirements Addressing the Objective	Rationale
	<p style="text-align: center;">156</p>	<p>of time those quota limitations are in place.</p> <p>FMT_MOF.1(4) provides the administrators “read only” access to the audit records and prohibits access to all other users. Additionally, the administrators are provided the capability to “search and sort” audit on defined criteria. This capability expedites problem resolution analysis.</p> <p>FMT_MOF.1(5) ensures that only an administrators can “enable or disable” the security alarms. This requirement works with FMT_MOF.1(5) to provide detailed granularity to the administrator when determining which actions constitute a security violation.</p> <p>FMT_MOF.1(6) provides the Security Administration configuration control of the allocation of connection-oriented TOE resources. This requirement provides the Security Administrator with a capability to thwart possible external “resource allocation” attacks on the TOE.</p> <p>FMT_MOF.1(7) provides the Security Administration configuration control of unsuccessful authentication attempts</p> <p>The requirement FMT_MTD.1(1) is intended to be used by the ST author, with possible iterations, to address TSF data that has not already been specified by other FMT requirements. This is necessary because the ST author may add TSF data in assignments that cannot be addressed ahead of time by the PP authors. This</p>

Objective	Requirements Addressing the Objective	Rationale
		<p>requirement specifies that the manipulation of these data be restricted to the security administrator.</p> <p>FMT_MTD.1(2) provides the Cryptographic Administrator, and only the Cryptographic Administrator, the ability to modify the cryptographic security data. This allows the Cryptographic Administrator to change the critical data that affects the TOE's ability to perform its cryptographic functions properly.</p> <p>FMT_MTD.1(3) provides the capability of setting the date and time that is used to generate time stamps to the Security Administrator or a trusted IT entity (authorized data manager). It is important to allow this functionality, due to clock drift and other circumstances, but the capability must be restricted. A trusted IT entity is allowed in the selection made by the ST author to take in account the use of an NTP server or some other service that provides time information without human intervention.</p> <p>FMT_MTD.1(4) addresses the capabilities of data managers, who have responsibilities for security data management for sub-portions of the set of TSF data (for example, the platform clock time, sub-hierarchies</p>

Objective	Requirements Addressing the Objective	Rationale
		<p>of the directory). The scope of a data manager's responsibility is set by a security administrator, but they are expected to manage the entities in their scope of control without reliance on the security administrator.</p> <p>FMT_MTD.2(1), FMT_MTD.2(2), FMT_MTD.2(3) restrict the setting of limits on the processor time, network connection resources and audit storage limits respectively, to an administrator. This capability allows an administrator to control the resources consumed by, to provide a flexible policy with respect to denial of service attacks.</p> <p>The requirement FMT_SMF.1 was introduced as an international interpretation. This requirement specifies functionality that must be provided to administrators of the TOE. If the PP author includes this requirement, care must be taken to use the other FMT requirements to specify how the functionality is restricted and to which role the functionality is provided.</p>

Objective	Requirements Addressing the Objective	Rationale
<p>O.MEDIATE_INFORMATION_FLOW</p> <p>The TOE must mediate the flow of information between sets of TOE network interfaces or between a network interface and the TOE itself in accordance with its security policy.</p>	<p>FDP_IFC.1(1)  FDP_IFC.1(2)  FDP_IFF.1(1)  FDP_IFF.1(2)  FMT_REV.1</p>	<p>The FDP_IFF and FDP_IFC requirements were chosen to define the policies, the subjects, objects, and operations for how and when mediation takes place.</p> <p>FDP_IFC.1(1), and FDP_IFC.1(2) define the subjects, information (e.g., objects) and the operations that are performed with respect to the two information flow policies.</p> <p>FDP_IFC.1(1) defines subjects for the unauthenticated access to any services the TOE provides. This is different from the other policies in that the TOE mediates access to itself, rather than determining if information should be allowed to flow through the TOE. The destination subject is defined to be the TOE, and the source subject is the TOE interface on which a network packet is received. The information remains the same, a network packet, and the operations are limited to accept or reject the packet.</p> <p>FDP_IFF.1(1) provides the rules that apply to the unauthenticated use of any services provided by the TOE. ICMP is the only service that is required to be provided by the TOE, and the security attributes associated with this protocol allow the Security Administrator to specify what degree the ICMP traffic is mediated (i.e., the ICMP message type and code).</p>

Objective	Requirements Addressing the Objective	Rationale
		<p>The ST author could specify other services they wish their TOE implementation to provide, and if they do so, they should also specify the security attributes associated with the additional services. FMT_REV.1 is a management requirement that affords the Security Administrator the ability to immediately revoke user's ability to send network traffic to or through the TOE.</p> <p>FDP_IFC.1(2), the subjects are the TOE's network interfaces. The objects are defined as the network IP packets on which the TOE performs routing operations. As packets enter the TOE, the network interface where they are received is the source subject. As packets are sent out of the TOE the network interface that they are sent out of is the destination subject. Subjects must be defined as entities that the TOE has control over. The TOE has control over its own network interfaces such that it can make information flow decisions to allow/disallow network packets to flow from in incoming interface to an outgoing interface, and can apply routing operations to packets that are allowed to flow. To define subjects as the senders and receivers of network packets would not allow specification of an information flow policy that the TOE could enforce, since the sender</p>

<b>Objective</b>	<b>Requirements Addressing the Objective</b>	<b>Rationale</b>
		<p>and receiver of network packets are not under the control of the TOE. The operations defined are those of the routing policy. The routing policy either passes information along or denies the information flow.</p> <p>FDP_IFF.1(2) specifies the attributes on which authenticated information flow decisions are made. Each TOE interface has a set of source subject identifiers that is the list of senders of information packets that are allowed to send packets to this TOE interface. Each TOE interface also has a list of destination subject identifiers that specifies the receivers that network packets can be sent to on that TOE interface. As packets are received on a particular network interface, the TOE determines if they are allowed to enter on that interface. Then based on rules defined by the Security Administrator, the TOE applies authenticated routing operations to the packet. Before the packet is sent out of a particular network interface, the TOE determines if the destination (i.e., receiver) of the packet is in the list of destinations that may be reached over that interface.</p> <p>FMT_REV.1 is a management requirement that affords the Security Administrator the ability to immediately revoke user's ability to send network traffic to or through</p>

Objective	Requirements Addressing the Objective	Rationale
		<p>the TOE.</p> <p>If the Security Administrator revokes a user's access (e.g., via a rule in the ruleset, revoking an administrative role from a user) the TOE will immediately enforce the new Security Administrator defined "policy". ADV_ARC.1 describes an architecture that ensures packets that flow through the TOE, or those that are destined for the TOE are mediated with respect to the identified policies. Each TSF interface that operates on subjects or objects that are identified in the explicit policies, or operates on TSF data or security attributes, must ensure that the operation is checked against the explicit and implicit security policies defined in this PP. If any TSF interface allows unchecked access to any of these resources, then the TOE cannot be relied upon to enforce the security policies.</p>

Objective	Requirements Addressing the Objective	Rationale
<p>O.PEER_AUTHENTICATI ON</p>	<p>FCS_IKE_(EXT).1</p>	<p>The O.PEER_AUTHENTICATI ON objective is satisfied by the requirement FCS_IKE_(EXT).1, which specifies that the TOE must implement the Internet Key Exchange protocol defined in RFC 2409. By implementing this protocol, the TOE will establish a secure, authenticated channel with each peer TOE for purposes of establishing a security association, which includes the establishment of a cryptographic key, algorithm and mode to be used for all communication. It is possible to establish multiple security associations between two peer TOEs, each with its own cryptographic key. Authentication may be via a digital signature or pre-shared key.</p>
<p>O.PROTECT_IN_TRANSI T</p> <p>The TSF shall protect TSF data when it is in transit between the TSF and another trusted IT entity.</p>	<p>FPT_ITA.1 FPT_ITC.1 FPT_ITI.1 FTP_ITC.1(1) FTP_ITC.1(2) FTP_TRP.1(1) FTP_TRP.1(2)</p>	<p>FPT_ITA.1, FPT_ITC.1 and FPT_ITI.1 are concerned with the availability, confidentiality and integrity of the TSF data while being transmitted.</p> <p>FTP_ITC.1(1) and FTP_ITC.1(2) ensures that all TSF data will be protected from disclosure while in transit from the TOE to another trusted IT entity.</p> <p>FTP_TRP.1(1) and FTP_TRP.1(2) will use cryptographic means to provide prevention of disclosure and detection of modification of TSF data.</p>

Objective	Requirements Addressing the Objective	Rationale
<p><b>O.PROTOCOLS</b></p> <p>The TOE will ensure that standardized protocols are implemented in the TOE to RFC and/or Industry specifications to ensure interoperability.</p>	<p>FPT_FLS.1</p> <p>FPT_PRO_(EXT).1</p>	<p>The O.PROTOCOLS objective is satisfied by FPT_PRO_(EXT).1, which requires that the TOE be implemented with standardized protocols to ensure interoperability among peer TOEs. Implementing the standardized protocols will ensure that a secure state (FPT_FLS.1) of the TOE is maintained.</p>
<p><b>O.REPLAY_DETECTION</b></p> <p>The TOE will provide a means to detect and reject the replay of TSF data and security attributes.</p>	<p>FPT_RPL.1</p>	<p>The O.REPLAY_DETECTION objective is satisfied by FPT_RPL.1, which requires the TOE to detect and reject the attempted replay of authentication data from a remote user. This is sufficient to meet the objective because no untrusted users have local access to the TOE, thus there is no way to capture neither replay authentication data for a local session.</p>

Objective	Requirements Addressing the Objective	Rationale
<p>O.RESIDUAL_INFORMATION</p> <p>The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated.</p>	<p>FDP_RIP.2</p> <p>FCS_CKM.4</p> <p>FCS_CKM_(EXT).2</p>	<p>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. For this TOE it is critical that the memory used to build network packets is either cleared or that some buffer management scheme be employed to prevent the contents of a packet being disclosed in a subsequent packet (e.g., if padding is used in the construction of a packet, it must not contain another user's data or TSF data).</p> <p>FCS_CKM.4 applies to the destruction of cryptographic keys used by the TSF. This requirement specifies how and when cryptographic keys must be destroyed. The proper destruction of these keys is critical in ensuring the content of these keys cannot possibly be disclosed when a resource is reallocated to a user.</p>

Objective	Requirements Addressing the Objective	Rationale
<p>O.RESOURCE_SHARING</p> <p>The TOE shall provide mechanisms that mitigate attempts to exhaust connection-oriented resources provided by the TOE (e.g., entries in a connection state table; Transmission Control Protocol (TCP) connections to the TOE).</p>	<p>FRU_RSA.1(1)</p> <p>FRU_RSA.1(2)</p> <p>FMT_MTD.2(1)</p> <p>FMT_MTD.2(2)</p> <p>FMT_MOF.1 (6)</p>	<p>While an availability security policy does not explicitly exist, FRU_RSA.1 was used to mitigate potential resource exhaustion attempts. FRU_RSA.1(1) was used to reduce the impact of an attempt being made to exhaust the transport-layer representation (e.g., attempt to make the TSF unable to respond to connection-oriented requests, such as SYN attacks). This requirement allows the administrator to specify the time period in which when maximum quota (which is defined by the ST) is met or surpassed, an ST defined action is to take place, which is specified in FMT_MTD.2(1). These two requirements together help limit the resources that can be utilized by the general population of users as a whole. An issue with treating all the users the same is that legitimate users may not be able to establish connections due to the connection table entries being exhausted. Therefore FRU_RSA.1(2) is also included.</p> <p>FRU_RSA.1(2) is more specific in that attempts to exhaust the connection-oriented resources by a single network address, or a set of network addresses can be controlled. This affords the administrator a finer granularity of control than FRU_RSA.1(1).</p>

Objective	Requirements Addressing the Objective	Rationale
		<p>FRU_RSA.1(2) has the advantage of providing the Security Administrator with the ability to define the maximum number of resources a particular address or set of addresses can use over a specified time period. This requirement works in conjunction with FMT_MTD.2(2) which restricts the ability to set the quotas to the security administrator and allows for the ST author to assign what actions will take place once the quotas are met or surpassed. This iteration of FPT_RSA.1 makes it less likely that a legitimate user of the TOE will be denied access due to resource exhaustion attempts.</p> <p>FMT_MOF.1(6) restricts the ability to assign the single network address or set of network addresses used in FRU_RSA.1(2) to the Security Administrator. This is in keeping with the TOE's notion of the Security Administrator is responsible for configuring the TOE's policy enforcement mechanisms.</p>
<p><b>O.ROBUST_ADMIN_GUI DANCE</b></p> <p>The TOE will provide administrators with the necessary information for secure delivery and management.</p>	<p>ALC_DEL.1 AGD_PRE.1 AGD_OPE.1</p>	<p>ALC_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</p>

Objective	Requirements Addressing the Objective	Rationale
		<p>necessary for secure management of the TOE.</p> <p>The AGD_PRE.1 requirement 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 Operational Users Guidance and the Preparative procedures ensure that once the administrator has followed the installation and configuration guidance the result is a TOE in a secure configuration.</p> <p>The AGD_OPE.1 requirement 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 ruleset 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.</p> <p>The AGD_OPE.1 is also 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).</p>

Objective	Requirements Addressing the Objective	Rationale
<p>O.ROBUST_TOE_ACCESS</p> <p>The TOE will provide mechanisms that control a user's logical access to the TOE and to explicitly deny access to specific users when appropriate</p>	<p>FTA_TSE.1</p> <p>FIA_UID.2</p> <p>FTA_SSL.3</p> <p>FIA_AFL.1</p> <p>FIA_ATD.1(1)</p> <p>FIA_ATD.1(2)</p> <p>FIA_UAU.2</p> <p>FIA_UAU_(EXT).5</p>	<p>FIA_UID.2 plays a small role in satisfying this objective by ensuring that every user is identified before the TOE performs any mediated functions. FIA_ATD.1(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). This requirement allows a human user to have more than one user identity assigned, so that a single human user could assume all the roles necessary to manage the TOE. In order to ensure a separation of roles, this PP requires a single role to be associated with a user id. This is inconvenient in that the administrator would be required to log in with a different user id each time they wish to assume a different role, but this helps mitigate the risk that could occur if an administrator were to execute malicious code.</p> <p>FIA_ATD.1(2) defines the attributes of IT entities, including a subject ID that is used to by the TOE to determine an entity's identity and enforce what type of access the entity has to the TOE. This requirement allows an IT entity to have more than one subject identity assigned, so that a single entity (e.g., another router) could assume the necessary role required to manage the TOE (e.g updating the routing tables).</p>

Objective	Requirements Addressing the Objective	Rationale
		<p>FIA_UAU.2 requires that administrators and authorized IT entities authenticate themselves to the TOE before performing any TSF-mediated actions. In order to control logical access to the TOE an authentication mechanism is required. The extended requirement FIA_UAU_(EXT).5 mandates that the TOE provide a local authentication mechanism. This requirement also affords the ST author the opportunity to add additional authentication mechanisms (e.g., single-use, certificates) if they desire.</p> <p>Local authentication is required to ensure someone that has physical access to the TOE and has not been granted logical access (e.g., a janitor) cannot gain unauthorized logical access to the TOE.</p> <p>FTA_TSE.1.1 contributes to this objective by limiting a user's ability to logically access the TOE. This requirement provides the Security Administrator the ability to control when (e.g., time and day(s) of the week) and where (e.g., from a specific network address) remote administrators, as and authorized IT entities can access the TOE.</p> <p>FIA_AFL.1 provides a detection mechanism for unsuccessful authentication attempts by remote administrators, and authorized IT entities. The requirement enables a Security Administrator settable threshold that prevents unauthorized</p>

Objective	Requirements Addressing the Objective	Rationale
		<p>users from gaining access to authorized user's account by guessing authentication data by locking the targeted account until the Security Administrator takes some action (e.g., re-enables the account) or for some Security Administrator defined time period. Thus, limiting an unauthorized user's ability to gain unauthorized access to the TOE.</p> <p>The FTA_SSL family partially satisfies the O.ROBUST_TOE_ACCESS objective by ensuring that user's sessions are afforded some level of protection. FTA_SSL.3 takes into account remote sessions. After a Security Administrator defined time interval of inactivity remote sessions will be terminated. This includes user remote administrative sessions. This component is especially necessary; since remote sessions are not typically afforded the same physical protections those local sessions are provided.</p>

Objective	Requirements Addressing the Objective	Rationale
<p>O.SELF_PROTECTION</p> <p>The TSF will maintain a domain for its own execution that protects itself and its resources from external interference, tampering, or unauthorized disclosure.</p>	<p>ADV_ARC.1</p>	<p>ADV_ARC.1 will require an architecture that 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. ADV_ARC.1 also provides an archecture to ensure that the cryptographic module be provided its own address space. This is necessary to reduce the impact of programming errors in the remaining portions of the TSF on the cryptographic module.</p> <p>The inclusion of ADV_ARC.1 ensures that the TSF makes policy decisions on all interfaces that perform operations on subjects and objects that are scoped by 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, and what access control is provided within the interfaces.</p>

Objective	Requirements Addressing the Objective	Rationale
<p>O.SOUND_DESIGN</p> <p>The design of the TOE will be the result of sound design principles and techniques; the design of the TOE, and the design principles and techniques, are adequately and accurately documented.</p>	<p>ADV_ARC.1</p> <p>ADV_FSP.5</p> <p>ADV_TDS.4</p> <p>ADV_INT.3</p>	<p>There are two different perspectives for this objective. One is from the developer's point of view and the other is from the evaluator's. The ADV class of requirements is levied to aide in the understanding of the design for both parties, which ultimately helps to ensure the design is sound.</p> <p>ADV_ARC.1 addresses the non-bypassability and domain separation aspects of the TSF, since this needs to be analyzed differently from other requirements. The low-level design, as required by ADV_TDS.4, provides the reader with the details of the TOE's design and describes at a module level how the design of the TOE addresses the SFRs. This level of description provides the detail of how modules interact within the TOE and if a flaw exists in the TOE's design, it is more likely to be found here rather than the high-level design. This requirement also mandates that the interfaces presented by modules be specified. Having knowledge of the parameters a module accepts, the errors that can be returned and a description of how the module works to support the security policies allows the design to be understood at its lowest level.</p>

Objective	Requirements Addressing the Objective	Rationale
		<p>ADV_INT.3 ensures that the design of the TOE has been performed using good software engineering design principles that require a modular design of the TSF. Modular code increases the developer's understanding of the interactions within the TSF, which in turn, potentially reduces the amount of errors in the design. Having a modular design is imperative for evaluator's to gain an appropriate level of understanding of the TOE's design in a relatively short amount of time. The appropriate level of understanding is dictated by other assurance requirements in this PP (e.g., ATE_DPT.3, AVA_CCA_(EXT).1, AVA_VAN.4).</p>

Objective	Requirements Addressing the Objective	Rationale
		<p>ADV_FSP.5 requires that the interfaces to the TSF be completely specified. In this TOE, a complete specification of the network interface (including the network interface card) is critical in understanding what functionality is presented to untrusted users and how that functionality fits into the enforcement of security policies. Some network protocols have inherent flaws and users have the ability to provide the TOE with network packets crafted to take advantage of these flaws. The routines/functions that process the fields in the network protocols allowed (e.g., TCP, UPD, ICMP, any application level) must fully specified: the acceptable parameters, the errors that can be generated, and what, if any, exceptions exist in the processing. The functional specification of the hardware interface (e.g., network interface card) is also extremely critical. Any processing that is externally visible performed by NIC must be specified in the functional specification. Having a complete understanding of what is available at the TSF interface allows one to analyze this functionality in the context of design flaws.</p>

Objective	Requirements Addressing the Objective	Rationale
		<p>ADV_TDS.4 requires that design of the TOE be provided. This level of design describes the architecture of the TOE in terms of subsystems. It identifies which subsystems are responsible for making and enforcing security relevant (e.g., anything relating to an SFR) decisions and provides a description, at a high level, of how those decisions are made and enforced. Having this level of description helps provide a general understanding of how the TOE works, without getting buried in details, and may allow the reader to discover flaws in the design.</p> <p>ADV_TDS.4 also provides the reader with the details of the TOE's design and describes at a module level how the design of the TOE addresses the SFRs. This level of description provides the detail of how modules interact within the TOE and if a flaw exists in the TOE's design, it is more likely to be found here rather than the high-level design. This requirement also mandates that the interfaces presented by modules be specified. Having knowledge of the parameters a module accepts, the errors that can be returned and a description of how the module works to support the security policies allows the design to be understood at its lowest level.</p>

Objective	Requirements Addressing the Objective	Rationale
<p>O.SOUND_IMPLEMENTATION</p> <p>The implementation of the TOE will be an accurate instantiation of its design, and is adequately and accurately documented.</p>	<p>ADV_IMP.1  ADV_ARC.1  ADV_TDS.4  ADV_INT.3  ALC_TAT.1</p>	<p>While ADV_TDS.4 is used to aide in ensuring that the TOE's design is sound, it also contributes to ensuring the implementation is correctly realized from the design. It is expected that evaluators will use the low-level design as an aide in understanding the implementation representation. The low-level design requirements ensure the evaluators have enough information to intelligently analyze (e.g., the documented interface descriptions of the modules match the entry points in the module, error codes returned by the functions in the module are consistent with those identified in the documentation) the implementation and ensure it is consistent with the design.</p> <p>ADV_IMP.2 was chosen to ensure evaluators have full access to the source code. If the evaluators are limited in their ability to analyze source code they may not be able to determine the accuracy of the implementation or the adequacy of the documentation. Often times it is difficult for an evaluator to identify the complete sample of code they wish to analyze.</p>

Objective	Requirements Addressing the Objective	Rationale
		<p>Often times looking at code in one subsystem may lead the evaluator to discover code they should look at in another subsystem. Rather than require the evaluator to “re-negotiate” another sample of code, the complete implementation representation is required.</p> <p>When performing the activities associated with the ADV_INT.3 requirement, the evaluators will ensure that the architecture of the implementation is modular and consistent with the architecture presented in the low-level design. Having a modular implementation provides the evaluators with the ability to more easily assess the accuracy of the implementation, with respect to the design. If the implementation is overly complex (e.g., circular dependencies, not well understood coupling, reliance on side-effects) the evaluator may not have the ability to assess the accuracy of the implementation.</p> <p>ALC_TAT.1 provides evaluators with information necessary to understand the implementation representation and what the resulting implementation will consist of. Critical areas (e.g., the use of libraries, what definitions are used, compiler options) are documented so the evaluator can determine how the implementation representation is to be analyzed.</p>

Objective	Requirements Addressing the Objective	Rationale
<p>O.THOROUGH_FUNCTIONAL_TESTING</p> <p>The TOE will undergo appropriate security functional testing that demonstrates the TSF satisfies the security functional requirements.</p>	<p>ATE_COV.3</p> <p>ATE_FUN.1</p> <p>ATE_DPT.3</p> <p>ATE_IND.2</p>	<p>In order to satisfy O.FUNCTIONAL_TESTING, the ATE class of requirements is necessary. The component ATE_FUN.1 requires the developer to provide the necessary test documentation to allow for an independent analysis of the developer's security functional test coverage. In addition, the developer must provide the test suite executables and source code, which are used for independently verifying the test suite results and in support of the test coverage analysis activities. ATE_COV.3 requires the developer to provide a test coverage analysis that demonstrates the TSFI are completely addressed by the developer's test suite. While exhaustive testing of the TSFI is not required, this component ensures that the security functionality of each TSFI is addressed. This component also requires an independent confirmation of the completeness of the test suite, which aids in ensuring that correct security relevant functionality of a TSFI is demonstrated through the testing effort. ATE_DPT.3 requires the developer to provide a test coverage analysis that demonstrates depth of coverage of the test suite.</p>

Objective	Requirements Addressing the Objective	Rationale
		<p>This component complements ATE_COV.3 by ensuring that the developer takes into account the high-level and low-level design when developing their test suite. Since exhaustive testing of the TSFI is not required, ATE_DPT.3 ensures that subtleties in TSF behavior that are not readily apparent in the functional specification are addressed in the test suite.</p> <p>ATE_IND.2 requires an independent confirmation of the developer's test results, by mandating a subset of the test suite be run by an independent party. This component also requires an independent party to attempt to craft functional tests that address functional behavior that is not demonstrated in the developer's test suite. Upon successful adherence to these requirements, the TOE's conformance to the specified security functional requirements will have been demonstrated.</p>

<b>Objective</b>	<b>Requirements Addressing the Objective</b>	<b>Rationale</b>
<p>O.TIME_STAMPS</p> <p>The TOE shall provide reliable time stamps and the capability for the administrator to set the time used for these time stamps.</p>	<p>FPT_STM.1</p> <p>FMT_MTD.1(3)</p>	<p>FPT_STM.1 requires that the TOE be able to provide reliable time stamps for its own use and therefore, partially satisfies this objective. Time stamps include date and time and are reliable in that they are always available to the TOE, and the clock must be monotonically increasing.</p> <p>FMT_MTD.1(3) satisfies the rest of this objective by providing the capability to set the time used for generating time stamps to either the Security Administrator, authorized IT entity, or both, depending on the selection made by the ST author.</p>

Objective	Requirements Addressing the Objective	Rationale
<p>O.TRUSTED_PATH</p> <p>The TOE will provide a means to ensure users are not communicating with some other entity pretending to be the TOE, and that the TOE is communicating with an authorized IT entity and not some other entity pretending to be an authorized IT entity.</p>	<p>FTP_ITC.1(1), FTP_ITC.1(2)</p> <p>FTP_TRP.1(1), FTP_TRP.1(2)</p>	<p>FTP_TRP.1.1 requires the TOE to provide a mechanism that creates a distinct communication path that protects the data that traverses this path from disclosure or modification. This requirement ensures that the TOE can identify the end points and ensures that a user cannot insert themselves between the user and the TOE, by requiring that the means used for invoking the communication path cannot be intercepted and allow a “man-in-the-middle-attack” (this does not prevent someone from capturing the traffic and replaying it at a later time – see FPT_RPL.1). Since the user invokes the trusted path (FTP_TRP.1.2) mechanism they can be assured they are communicating with the TOE. FTP_TRP.1.3 mandates that the trusted path be the only means available for providing identification and authentication information, therefore ensuring a user’s authentication data will not be compromised when performing authentication functions. Furthermore, the remote administrator’s communication path is encrypted during the entire session.</p>

Objective	Requirements Addressing the Objective	Rationale
		<p>FTP_ITC.1(1) and FTP_ITC.1(2) are similar to FTP_TRP.1(1) and FTP_TRP.1(2), in that they require a mechanism that creates a distinct communication path with the same characteristics, however FTP_ITC.1(1) and FTP_ITC.1(2) is used to protect communications between IT entities, rather than between a human user and an IT entity. FTP_ITC.1.3 requires the TOE to initiate the trusted channel, which ensures that the TOE has established a communication path with an authorized IT entity and not some other entity pretending to be an authorized IT entity.</p>
<p><b>O.USER_GUIDANCE</b> The TOE will provide users with the information necessary to correctly use the security mechanisms.</p>	<p><b>AGD_PRE.1</b></p>	<p><b>O.USER_GUIDANCE</b> (AGD_PRE.1) mitigates this threat by providing the user the information necessary to use the security mechanisms that control access to user data in a secure manner. For instance, the method by which the discretionary access control mechanism (FDP_ACC.1, FDP_ACF.1) is configured, and how to apply it to the data the user owns, is described in the user guidance. If this information were not available to the user, the information may be left unprotected, or the user may mis-configure the controls and unintentionally allow unauthorized access to their data.</p>

Objective	Requirements Addressing the Objective	Rationale
<p>O.VULNERABILITY_ANALYSIS_TEST</p> <p>The TOE will undergo appropriate independent vulnerability analysis and penetration testing to demonstrate the design and implementation of the TOE does not allow attackers with medium attack potential to violate the TOE's security policies.</p>	<p>AVA_VAN.4</p>	<p>To maintain consistency with the overall assurance goals of this TOE, O.VULNERABILITY_ANALYSIS_TEST requires the AVA_VAN.4 component to provide the necessary level of confidence that vulnerabilities do not exist in the TOE that could cause the security policies to be violated. AVA_VAN.4 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.</p>

## 6.4 Rationale for Assurance Requirements

153 The Evaluation Assurance Level (EAL) definitions and assurance requirements in Part 3 of the CC were used as a basis for inclusion in Medium Robustness Protection Profile Assurance Requirements. Section 5.3 was believed to best achieve the goal of addressing circumstances where developers and users require a moderate level of independently assured security in commercial products. This collection of assurance requirements require TOE developers to gain assurance from high-quality software engineering development practices which, though rigorous, do not require substantial specialist knowledge, skills, and other resources. Rationale for individual assurance requirements is provided in Table 8 above.

## 6.5 Rationale for Satisfying all Dependencies

**Table 9 Functional Requirement Dependencies**

Requirement	Dependency	Satisfied
FCS_CKM.1 FCS_CKM.4	FMT_MSA.2	This dependency is satisfied by including FMT_MSA.2.
FIA_UAU.1 FIA_UAU.2 FMT_SMR.2	FIA_UID.1	This dependency is satisfied with the inclusion of requirement FIA_UID.2. This requirement is hierarchical to FIA_UID.1 and is sufficient to satisfy the dependency for these requirements.
FMT_MOF.1 FMT_MSA.1 FMT_MTD.2 FMT_REV.1	FMT_SMR.1	This dependency is satisfied with the inclusion of requirement FMT_SMR.2. This requirement is hierarchical to FMT_SMR.1 and is sufficient to satisfy the dependency for these requirements.
FPT_RCV.2	FPT_TST.1	This dependency is satisfied with the inclusion of requirement FPT_TST_(EXT).1 which meets the dependency for self-test in FPT_RCV.2.

## 6.6 Rationale for Extended Requirements

154 Table 10 presents the rationale for the inclusion of the extended functional and assurance requirements found in this PP. The extended requirements that are included as NIAP interpretations do not require a rationale for their inclusion per CCEVS management. The extended requirements were generated to provide the level of assurance appropriate for the medium robustness environments.

Table 10 Rationale for Extended Requirements

Extended Requirement	Identifier	Rationale
FAU_ARP_ACK_(EXT).1	Security alarm acknowledgement	This extended requirement is necessary since a CC requirement does not exist to ensure an administrator will be aware of the alarm. The intent is to ensure that if an administrator is logged in and not physically at the console or remote workstation the message will remain displayed until the administrators have acknowledged it. The message will not be scrolled off the screen due to other activity-taking place (e.g., the auditor is running an audit report).
FCS_BCM_(EXT).1	Baseline cryptographic module	This extended requirement is necessary since the CC does not provide a means to specify a cryptographic baseline of implementation.
FCS_CKM_(EXT).2	Cryptographic key validation and storage	This extended requirement is necessary since the CC does not provide a means to specify a cryptographic key validation and storage method.
FCS_COP_(EXT).1	Random number generation	This extended requirement is necessary since the CC does not provide a means to perform random number generation. This service is specified in NIST Special Publication 800-22.

Extended Requirement	Identifier	Rationale
FCS_IKE_(EXT).1	Internet Key Exchange	This extended requirement is necessary since the CC does not include requirements for this specific key exchange protocol. This protocol is specified in RFC 2409, but there are specific configurable settings that must be specified that are documented in the extended requirement.
FIA_UAU_(EXT).5	Multiple authentication mechanisms	This extended requirement is needed for local administrators because there is no CC requirement that requires the TSF provide authentication. Because this PP allows the IT environment to provide an authentication server to be used for the single-use authentication mechanism for remote users, it is important to specify that the TSF provide the means for local administrator authentication in case the TOE cannot communicate with the authentication server.
FPT_PRO_(EXT).1	Standard protocol usage	This extended requirement is necessary since the CC does not provide requirements of choosing a standard protocol mechanism from the standard protocols being used by a particular IT product.

Extended Requirement	Identifier	Rationale
FPT_TST_(EXT).1	TSF testing	This extended requirement is necessary to capture the notion of the TOE to verify the integrity of the TSF software. Additionally, the TSF data set that is subject to these tests was reduced to address the notion that it does not make sense to test the integrity of some TSF data (e.g., audit data) and this extended requirement address that.
AVA_CCA_(EXT).1	Systematic cryptographic module covert channel analysis	This extended assurance requirement is deemed necessary to be performed on the entire TOE to determine the bandwidth of possible cryptographic key leakage in order to reduce the damage due to a key compromise.

## 7 APPENDICES

---

155 Section 7 of this document contains the appendices, that accompany the PP and provides clarity and/or explanation for the reader.

## A REFERENCES

---

- [1] Common Criteria for Information Technology Security Evaluation, *CCIMB-99-031, Version 2.2, January 2004.*
- [1a] Common Criteria for Information Technology Security Evaluation, *CCMB-2006-09, Version 3.1, September 2006.*
- [2] U.S. Government Protection Profile for Single-Level Operating Systems in Environments Requiring Medium Robustness, *Version 1.67, 30 October 2003*
- [2a] U.S. Government Protection Profile for Single-Level Operating Systems in Environments Requiring Medium Robustness, *Version 1.91, 16 March 2007*
- [3] *Department of Defense Chief Information Officer Guidance and Policy Memorandum No.6-8510, Guidance and Policy for the Department of Defense Global Information Grid Information Assurance (GIG), June 2000.*
- [3a] Department of Defense Directive 8500.1, "Information Assurance," October 24, 2002
- [3b] Department of Defense Instruction 8500.2, "Information Assurance," February 6, 2003
- [4] U.S. Department of Defense Virtual Private Network (VPN) Boundary Gateway Protection Profile for Basic Robustness Environments (BRE), *Version .6, September 2001.*
- [5] U.S. Government Firewall Protection Profile for Medium Robustness Environments, *Version 1.0, October 28, 2003.*
- [6] Federal Information Processing Standard Publication (FIPS-PUB) 197, Advanced Encryption Standard (AES), November 2001.

- [7] Federal Information Processing Standard Publication (FIPS-PUB) 140-2, Security Requirements for Cryptographic Modules, May 25, 2001.
- [8] Federal Information Processing Standard (FIPS-PUB) 46-3, Data Encryption Standard (DES), October 1999.
- [9] Internet Engineering Task Force, IP Encapsulating Security Payload (ESP), RFC 2406, November 1998.
- [10] Internet Engineering Task Force, Internet Key Exchange (IKE), RFC 2409, November 1998.
- [11] Information Assurance Technical Framework, *Release 3.0, September 2000.*
  - [11a] Information Assurance Technical Framework, *Release 3.1, September 2002.*
- [12] Internet Engineering Task Force, ESP CBC-Mode Cipher Algorithms, RFC 2451, November 1998.
- [13] Internet Engineering Task Force, Use of HMAC-SHA-1-96 within ESP and AH, RFC 2404, November 1998.
- [14] Department of Defense Instruction, Information Assurance Implementation, 8500.2, February 6, 2003.
- [15] Department of Defense Directive, Information Assurance, 8500.1, October 24, 2002.
- [16] Implementing Virtual Private Networks, Steven Brown, McGraw Hill, 1999.
- [17] A Goal VPN Protection Profile for Protecting Sensitive Information, *Release 2.0, July 2000.*
- [18] The AES Cipher Algorithm and Its Use with Ipsec <draft-ietf-ipsec-ciph-aes-cbc.03.txt>, *Internet draft, November 2001.*

## B GLOSSARY

---

**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<sup>2</sup> and the disclosure and modification of data.<sup>3</sup>

**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 the entire 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).

**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.

---

<sup>2</sup> Hardware and software.

<sup>3</sup> Stored or communicated.

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

**Availability** – Timely<sup>4</sup>, 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:

- the transformation of plaintext data into ciphertext data,
- the transformation of ciphertext data into plaintext data,
- a digital signature computed from data,
- the verification of a digital signature computed from data, or
- a digital authentication code computed from data.

---

<sup>4</sup> According to a defined metric.

***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. Those controls are discretionary in the sense that a subject with 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 OTE 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.<sup>5</sup>

***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 require 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 require 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.

---

<sup>5</sup> The Bell LaPadula model is an example of Mandatory Access Control.

**Operating System (OS)** – An entity within the TSC that causes operations to 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.

**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 operation key variable. The combining process excludes concatenation or interleaving of component variables.

***Subject*** – An entity within the TSC that causes operation 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.

## C ACRONYMS

---

ANSI	American National Standards Institute
BGP	Border Gateway Protocol
CC	Common Criteria
CCEVS	Common Criteria Evaluation Validation Scheme
CCIMB	Common Criteria Interpretations Management Board
CCM	Counter with Cipher Block Chaining-Message Authentication Code
CLNP	Connectionless Network Protocol
CLNP	Connectionless Network Protocol
CLNS	Connectionless Network Service
CM	Configuration Management
CSP	Cryptographic security parameter
DoD	Department of Defense
EAL	Evaluation Assurance Level
FIPS	Federal Information Processing Standard
FTP	File Transfer Protocol
GIG	Global Information Grid
HMAC	Keyed-Hash Authentication Code
HTTP	Hypertext Transfer Protocol
IATF	Information Assurance Technical Framework
ICMP	Internet Control Message Protocol

## Router Protection Profile For Medium Robustness Environments

IKE	Internet Key Exchange
IPSEC	Internet Protocol Security
IPv6	Internet Protocol Version 6
IPX	Internetwork Packet Exchange
IPX	Internetwork Packet Exchange
ISAKMP	Internet Security Association and Key Management Protocol
IS-IS	Intermediate System-to-Intermediate System
ISO	International Organization for Standardization
IT	Information Technology
LDP	Label Distribution Protocol
MAC	Mandatory Access Control
MPLS	Multi-protocol Label Switching
NIAP	National Information Assurance Partnership
NIST	National Institute of Standards Technology
NSA	National Security Agency
NTP	Network Time Protocol
OSI	Open Systems Interconnect
OSPF	Open Shortest Path First
PKI	Public Key Infrastructure
PP	Protection Profile
PRNG	Prime Random Number Generator
RFC	Request for Comments

## Router Protection Profile For Medium Robustness Environments

RIP	Routing Information Protocol
RNG	Random Number Generator
RSA	Rivest, Shamir, Adelman
SA	Security Association
SFP	Security Functional Policy
SFR	Security Functional Requirement
SHA	Secure Hash Algorithm
ST	Security Target
TBD	To Be Determined
TCP/IP	Transmissions Control Protocol/ Internet Protocol
TDEA	Triple Data Encryption Algorithm
TFTP	Trivial File Transfer Protocol
TOE	Target of Evaluation
TSC	TOE Scope of Control
TSE	TOE Security Environment
TSF	TOE Security Functions
TSFI	TSF interfaces
TSP	TOE Security Policy
TTAP/CCEVS	Trust Technology Assessment Program/ Common Criteria Evaluation Standard Scheme
UDP	User Datagram Protocol
VPN	Virtual Private Network

## **D ROBUSTNESS ENVIRONMENT CHARACTERIZATION**

---

### **D.1 General Environmental Characterization**

- 156 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.
- 157 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).
- 158 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, these two environmental factors will be related to the robustness required for selection of an appropriate TOE.

#### **D.1.1 Value of Resources**

- 159 Value of the resources associated with the TOE includes the data being processed or used by the TOE, and 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.

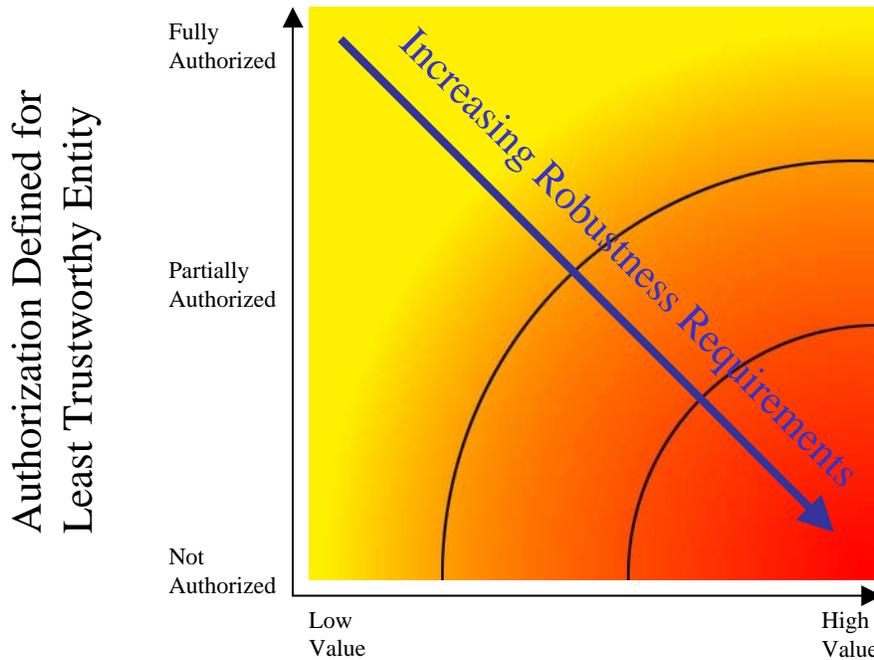
#### **D.1.2 Authorization of Entities**

- 160 Authorization that entities (users, administrators, and 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).
- 161 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.
- 162 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.

### D.1.3 Selection of Appropriate Robustness Levels

- 163 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.
- 164 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

- 165 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:
- 166 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.
- 167 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.
- 168 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.
- 169 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



Highest Value of Resources  
Associated with the TOE

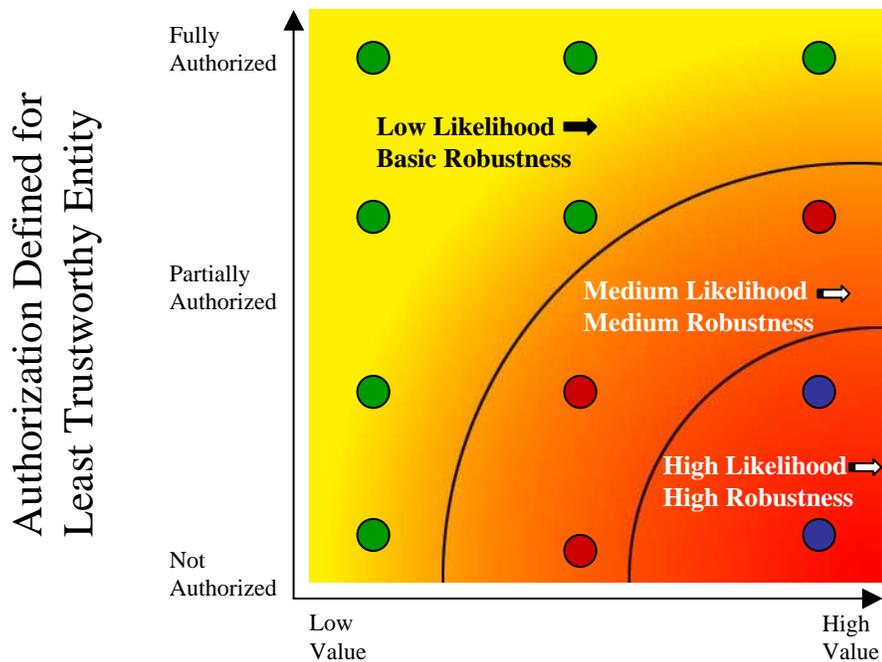
of the chart is intended to reflect- the notion that different environments engender similar levels of “likelihood of attempted compromise”, signified by a similar color. Further, the delineations between such environments are not stark, but rather are finely grained and gradual.

170 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 neither 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.

171 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 and 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.

- 172 The difficult part of this activity is differentiating the authorization of various entities, and 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 Section 3 of this PP, the targeted threat level for a medium robustness TOE is characterized. This information is provided to help organizations using this PP -ensure that the functional requirements specified by this medium robustness PP are appropriate for their intended application of a compliant TOE.



Highest Value of Resources Associated with the TOE

## E REFINEMENTS

---

173 This section contains refinements where text was omitted. Omitted text is shown as bold text within parenthesis. The actual text of the functional requirements as presented in Section 5 has been retained.

FAU\_ARP.1.1 – **Refinement:** The TSF shall (**take**) [immediately display an alarm message, identifying the potential security violation and make accessible the audit record contents associated with the auditable event(s) that generated the alarm, at the:

- local console,
- remote administrator sessions that exist, and;
- remote administrator sessions that are initiated before the alarm has been acknowledged, and;
- at the option of the Security Administrator, generate an audible alarm, and;
- [assignment: other methods]] upon detection of a potential security violation.

FAU\_GEN.1.1-NIAP-0410 – **Refinement:** The TSF shall be able to generate an audit record of the following auditable events:

- a) Start-up and shutdown of the audit functions;
- b) All auditable events (**for the**) [Table 4] (**level of audit; and**)
- 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 extended requirements determined by the ST Author], no additional events].

FAU\_GEN.1.2-NIAP-0410 – **Refinement:** 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, [(**selection**): [information specified in column three of Table 4 below]].

FAU\_GEN.2.1-NIAP-0410 – **Refinement:** (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.

FAU\_SAA.1.2-NIAP-0407 - **Refinement:** The TSF shall (**monitor**) **enforce the following rules for monitoring audited events:** a) accumulation or combination of [**the following events**:

- a) Security Administrator specified number of authentication failures;
- b) Security Administrator specified number of Information Flow policy violations by an individual presumed source network identifier (e.g., IP address) within an administrator specified time period;
- c) Security Administrator specified number of Information Flow policy violations to an individual destination network identifier within an administrator specified time period;
- d) Security Administrator specified number of Information Flow policy violations to an individual destination subject service identifier (e.g., TCP port) within an administrator specified time period;
- e) Security Administrator specified Information Flow policy rule, or group of rule violations within an administrator specified time period;
- f) Any detected replay of TSF data or security attributes;
- g) Any failure of the cryptomodule or key generation self-tests (FPT\_TST.1(1) and FPT\_TST.1(2));
- h) Any failure of the other TSF self-tests (FPT\_TST\_(EXT).1);
- i) Security Administrator specified number of encryption failures;
- j) Security Administrator specified number of decryption failures] known to indicate a potential security violation;
- k) [selection: [assignment: any other rules], "no additional rules"].

FAU\_STG.3.1 - **Refinement:** The TSF shall (**take**) [immediately alert the administrators by displaying a message at the local console, and at the remote administrative console when an administrative session exists for each of the defined administrative roles, at the option of the Security Administrator generate an audible alarm, [selection: [assignment: other methods determined by the ST Author], no other methods] if the audit trail exceeds [a Security Administrator settable percentage of storage capacity].

FAU\_STG.1.2-NIAP-0423 – **Refinement:** The TSF shall be able to *prevent* (**unauthorized**) modifications to the audit records in the audit trail.

FDP\_IFF.1.2(1) - **Refinement:** The TSF shall permit an information flow between a **source (controlled)** subject and a **destination subject (controlled information)** via a controlled operation if the following rules hold:

- [the presumed identity of the source subject is in the set of source subject identifiers;
- the identity of the destination subject is in the set of source destination identifiers;
- the information security attributes match the attributes in an information flow policy rule (contained in the information flow policy ruleset defined by the Security Administrator) according to the following algorithm [assignment: algorithm used by the TOE to match information security attributes to information flow policy rules]; and
- the selected information flow policy rule specifies that the information flow is to be permitted].

FDP\_IFF.1.2(2) - **Refinement:** The TSF shall permit an information flow between a **source** subject and a **destination subject** via a controlled operation if the following rules hold:

- [the source subject has successfully authenticated to the TOE;
- the identity of the destination subject is in the set of destination identifiers;
- the information security attributes match the attributes in a information flow policy rule (contained in the information flow policy ruleset defined by the administrator) according to the following algorithm [assignment: algorithm used by the TOE to match information security attributes to information flow policy rules]; and

- the selected information flow policy rule specifies that the information flow is to be permitted].

FIA\_AFL.1.2-NIAP-0425 – **Refinement:** When the defined number of unsuccessful authentication attempts has been met (**or surpassed**), the TSF shall [at the option of the Security Administrator prevent the remote administrators or authorized IT entity from performing activities that require authentication until an action is taken by the Security Administrator, or until a Security Administrator defined time period has elapsed].

FIA\_ATD.1.1(1) **Refinement:** The TSF shall maintain the following list of security attributes belonging to (**individual users**) **authorized user:**

a) [user identifier(s):

role;

[selection: [assignment: Any security attributes related to a user identifier (e.g., certificate associated with the userid)], none]; and

b) [selection: [assignment: other user security attributes], none]].

FIA\_ATD.1.1(2) **Refinement:** The TSF shall maintain the following list of security attributes belonging to (**individual users**) **authorized subjects:**

a) [subject identity;

b) [assignment: any other security attributes].

FMT\_MSA.3.1-NIAP-0409(1) – **Refinement:** The TSF shall enforce the [UNAUTHENTICATED INFORMATION FLOW SFP, AUTHENTICATED INFORMATION FLOW SFP] to provide restrictive default values for (**security attributes**) **the information flow policy ruleset** that (**are**) is used to enforce the SFP.

FMT\_MSA.3.1-NIAP-0409(2) – **Refinement:** The TSF shall enforce the [UNAUTHENTICATED TOE SERVICES SFP] to provide restrictive default values for (**security attributes**) (**that are used to enforce the SFP**) **the set of TOE services available to unauthenticated users.**

FMT\_REV.1.2 - **Refinement:** The TSF shall **immediately** enforce the (**rules**):

- [revocation of a user's role (Security Administrator, Cryptographic Administrator, Audit Administrator);
- changes to the information flow policy ruleset when applied;
- disabling of a service available to unauthenticated users; and
- [selection: [assignment: other rules as determined by the ST Author], none]].

FPT\_SEP.2.3 - **Refinement:** The TSF shall maintain the part of the TSF related to [cryptography] in (**security domain(s)**) **an address space** for (**their**) **its** own execution that protects (**them**) **it** from interference and tampering by the remainder of the TSF and by subjects untrusted with respect to (**those SFPs**) **the cryptographic functionality**.

FTA\_SSL.1.2 - **Refinement:** The TSF shall require the (**following events to occur**) **user to re-authenticate** prior to unlocking the session(: [assignment: events to occur]).

FTA\_SSL.2.2 - **Refinement:** The TSF shall require the (**following events to occur**) **user to re-authenticate** prior to unlocking the session(: [assignment: events to occur]).

FTA\_TSE.1.1 - **Refinement:** The TSF shall be able to deny (**session**) **establishment of an authorized user session** based on [location, time, and day].

FTP\_ITC.1.1(1) - **Refinement:** The TSF shall (**provide**) **use encryption to** provide a **trusted** communication channel between itself and (**a remote trusted IT product**) **authorized IT entities** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from (**modification or**) disclosure.

FTP\_ITC.1.1(2) - **Refinement:** The TSF shall (**provide**) **use a cryptographic signature** to provide a **trusted** communication channel between itself and (**a remote trusted IT product**) **authorized IT entities** that is logically distinct from other communication channels and provides assured identification of its end points and (**protection of the channel data from modification or disclosure**) **detection of the modification of data**.

FTP\_TRP.1.1(1) - **Refinement:** The TSF shall provide (**a**) **an encrypted** communication path between itself and remote **administrators and authenticated** users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from (**modification or**) disclosure.

FTP\_TRP.1.1(2) - **Refinement:** The TSF shall **use a cryptographic signature to** provide a communication path between itself and remote **administrators and authenticated** users that is logically distinct from other communication paths and provides assured identification of its end points and **(protection) detection (of the communicated data from) modification (or disclosure) of data.**

FTP\_ITC.1.1(1) - **Refinement:** The **(TSF) IT Environment** shall provide a **trusted** communication channel between itself and the **(a remote trusted IT product) TSF** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from **(modification or) disclosure.**

FTP\_ITC.1.1(2) - **Refinement:** The **(TSF) IT Environment** shall provide **(a) an encrypted** communication channel between itself and **(a remote trusted IT product) the TSF** that is logically distinct from other communication channels and provides assured identification of its end points and **(protection of the channel data from modification or disclosure) detection of the modification of data.**

FTP\_TRP.1.1(1) - **Refinement:** The **(TSF) IT Environment** shall provide **(a) an encrypted** communication path between itself and **([selection: remote, local]) the TSF (users)** that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from modification or disclosure.

FTP\_TRP.1.2(1) - The **(TSF) IT Environment** shall permit remote users **of the TSF** to initiate communication **to the TSF** via the trusted path.

FTP\_TRP.1.3(1) – **Refinement:** The **(TSF) IT Environment** shall **(require) initiate** the use of the trusted path for **(initial) user authentication, all remote administration actions**, [selection: [assignment: other services for which trusted path is required], none].

FTP\_TRP.1.1(2) - **Refinement:** The **(TSF) IT Environment** shall provide **(a) an encrypted** communication path between itself and **([selection: remote, local]) (users) the TSF** that is logically distinct from other communication paths and provides assured identification of its end points **(protection of the communicated data from modification or disclosure) and detection of the modification of data.**

FTP\_TRP.1.2(2) - The **(TSF) IT Environment** shall permit *remote users of the TSF* to initiate communication **to the TSF** via the trusted path.

FTP\_TRP.1.3(2) – **Refinement:** The **(TSF) IT Environment** shall **(require)** **initiate** the use of the trusted path for **(initial) user authentication, all remote administration actions**, [*selection: [assignment: other services for which trusted path is required], none*].

## F STATISTICAL RANDOM NUMBER GENERATOR TESTS

---

- 1 A cryptographic module employing random number generators (RNGs) shall perform the following statistical tests for randomness. A single bit stream of 20,000 consecutive bits of output from each RNG shall be subjected to the following four tests: monobit test, poker test, runs test, and long runs test. (These four tests are simply those that formerly existed as the statistical RNG tests in Federal Information Processing Standard 140-2. However, for purposes of meeting this protection profile, these tests must be performed at the frequency specified earlier in this protection profile.)

### The Monobit Test:

1. Count the number of ones in the 20,000 bit stream. Denote this quantity by X.
2. The test is passed if  $9,725 < X < 10,275$ .

### The Poker Test:

1. Divide the 20,000 bit stream into 5,000 contiguous 4 bit segments. Count and store the number of occurrences of the 16 possible 4 bit values. Denote  $f(i)$  as the number of each 4 bit value  $i$ , where  $0 < i < 15$ .
2. Evaluate the following:

$$X = (16 / 5000) * \left( \sum_{i=0}^{15} [f(i)]^2 \right) - 5000$$

3. The test is passed if  $2.16 < X < 46.17$ .

### The Runs Test:

1. A run is defined as a maximal sequence of consecutive bits of either all ones or all zeros that is part of the 20,000 bit sample stream. The incidences of runs (for both consecutive zeros and consecutive ones) of all lengths ( $> 1$ ) in the sample stream should be counted and stored.
2. The test is passed if the runs that occur (of lengths 1 through 6) are each within the corresponding interval specified in the table below. This must hold for both the zeros and ones (i.e., all 12 counts must lie in the specified interval). For the purposes of this test, runs of greater than 6 are considered to be of length 6.

**Table C.1 - Required Intervals for Length of Runs Test**

Length of Run	Required Interval
1	2343 - 2657

## Router Protection Profile For Medium Robustness Environments

2	1135 - 1365
3	542 - 708
4	251 - 373
5	111 - 201
6 and greater	111 - 201

### The Long Runs Test:

1. A long run is defined to be a run of length 26 or more (of either zeros or ones).
2. On the sample of 20,000 bits, the test is passed if there are no long runs.

## End Notes

---

- i A deletion of CC text was performed in FPT\_TST.1.1(1). Rationale: The word "TSF" was deleted to allow for the demonstration of the correct operation of a number of cryptographic related self tests.  
FPT\_TST.1.1(1) **Refinement:** The TSF shall run a suite of self-tests **in accordance with FIPS PUB 140-2, Level 4 (as identified in Table 5.3) during initial start-up (on power on), at the request of the cryptographic administrator (on demand), under various conditions, and periodically (at least once a day)** to demonstrate the correct operation of the ~~TSF~~ **following ...**
- ii A deletion of CC text was performed in FPT\_TST.1.2(2). Rationale: The word "users" was deleted to replace it with the role of "cryptographic administrator". "Only authorized cryptographic administrators should be given the capability to verify the integrity of cryptographically related TSF data."  
FPT\_TST.1.2(1) **Refinement:** The TSF shall provide authorized ~~users~~ **cryptographic administrators** with the capability to verify the integrity of **TSF data related to the cryptography by using TSF-provided cryptographic functions.**
- iii A deletion of CC text was performed in FPT\_TST.1.3(1). Rationale: The word "users" was deleted to replace it with the role of "cryptographic administrator". Only authorized cryptographic administrators should be given the capability to verify the integrity of cryptographically related TSF executable code.  
FPT\_TST.1.3(1) **Refinement:** The TSF shall provide authorized ~~users~~ **cryptographic administrators** with the capability to verify the integrity of stored **cryptographically related** TSF executable code.
- iv A deletion of CC text was performed in FPT\_TST.1.1(2). Rationale: The words "the TSF" was deleted to allow for the demonstration of the correct operation of each key generation component. The word "perform" replaced "run a suite of" for clarity and better flow of the requirement.  
FPT\_TST.1.1(2) **Refinement:** The TSF shall ~~run a suite of~~ **perform** self-tests **immediately after generation of a key** to demonstrate the correct operation of ~~the TSF~~ **each key generation component. If any of these tests fails, that generated key shall not be used, the cryptographic module shall react as required by FIPS PUB 140 for failing a self-test, and this event will be audited.**
- v A deletion of CC text was performed in FPT\_TST.1.2(2). Rationale: The word "users" was deleted to replace it with the role of "cryptographic administrator".  
FPT\_TST.1.2(2) **Refinement:** The TSF shall provide authorized ~~users~~ **cryptographic administrators** with the capability to verify the integrity of TSF data **related to the key generation.**
- vi A deletion of CC text was performed in FPT\_TST.1.3(2). Rationale: The word "users" was deleted to replace it with the role of "cryptographic administrator".  
FPT\_TST.1.3(2) **Refinement:** The TSF shall provide authorized ~~users~~ **cryptographic administrators** with the capability to verify the integrity of stored TSF executable code **related to the key generation.**