US Government

Wireless Local Area Network (WLAN) Client

Protection Profile

For

Basic Robustness Environments

July 25, 2007

Version 1.1

Protection Profile Title:

U.S. Government Protection Profile Wireless Local Area Network (WLAN) Client Basic Robustness Environments

Criteria Version:

This Protection Profile "*US Government Protection Profile Wireless Local Area Network (WLAN) Client for Basic Robustness Environments"* (PP) was updated using Version 3.1 of the Common Criteria (CC).

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.

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.

Any issue that may arise when claiming compliance with this PP can be resolved using the observation report (OR) and observation decision (OD) process.

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. 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

List of Tables and Figures

Conventions and Terminology

Conventions

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 Common Criteria (CC). Selected presentation choices are discussed here to aid the PP reader.

The notation, formatting, and conventions used in this PP are largely consistent with those used in version 3.1 of the Common Criteria (CC). Selected presentation choices are discussed here to aid the PP user.
The CC allows several operations to be performed on functional requirements; *refinement*, *selection*, *assignment*, and *iteration* are defined in paragraph C4 of Part 1 of the CC. Each of these operations is used in this PP.
The **refinement** operation is used to add detail to a requirement, and thus further restricts a requirement. Refinement of security requirements is denoted by **bold text**.
The **selection** operation is used to select one or more options provided by the CC in stating a requirement. Selections are denoted by *italicized text*.
The **assignment** operation is used to assign a specific value to an unspecified parameter, such as the length of a password. Assignment is indicated by showing the value in square brackets, [Assignment_value].
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 3component identifier, (iteration_number).
The **security target author** operation is used to denote points in which the final determination of attributes is left to the security target writer. Security target writer operations are indicated by the words “ST AUTHOR -”.
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.
Application Notes are provided to help the developer, either to clarify the intent of a requirement, identify implementation choices, or to define “pass-fail” criteria for a requirement. For those components where Application Notes are appropriate, the Application Notes will follow the requirement component.

NAMING CONVENTIONS

Assumptions: TOE security environment assumptions are given names beginning with “A.”—e.g., A.ADMINISTRATION.

Threats: TOE security environment threats are given names beginning with “T.”—e.g., T.SIGNAL_DETECT.

Policies: TOE security environment policies are given names beginning with “P.”—e.g., P.GUIDANCE.

Objectives: Security objectives for the TOE and the TOE environment are given names beginning with “O.” and “OE.”, respectively,—e.g., O.ACCESS and OE.ADMIN.

Terminology

In the CC 3.1, Section 4 of Part 1 defines many terms. In addition to terms defined in the CC, this PP references the following defined terms.

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[1] and the disclosure and modification of data.[2]

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

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

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 credentials -- Information used to verify a claimed identity.

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

Availability -- Timely[3], reliable access to IT resources.

Compromise -- Violation of a security policy.

[1] Hardware and software.
[2] Stored or communicated.
[3] According to a defined metric.

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 boundary -- An explicitly defined continuous perimeter that establishes the physical bounds of a cryptographic module and contains all the hardware, software, and/or firmware components of a cryptographic module.

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

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

Cryptographic Module -- The set of hardware, software, and/or firmware that implements FIPS Approved security functions (including cryptographic algorithms and key generation) and is contained within the cryptographic boundary.

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.

Cryptomodule – see cryptographic module.

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.

MAC Address -- Media Access Control Address, the globally unique 48 bit media layer address of a network device. Sometimes referred to as the physical address.

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

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

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.

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.

TOE Security Function (TSF) Data -- Information used by the TSF in making TOE security policy (TSP) decisions. TSF data may be influenced by users if allowed by the TSP. Security attributes, authentication data, and access control list entries are examples of TSF data.

Unauthorized User -- Any person who is not authorized, under the TSP, to access the TOE. This definition authorized users who seek to exceed their authority.

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.

Wireless Client -- A device consisting of hardware and software used to provide a wireless interface to communicate with other wireless devices.

Document Organization

Section 1 provides the introductory material for this PP. It includes an introduction, a brief description of the WLAN client TOE and additional identifying information. It also includes a discussion of the factors used to define the TOE environment and the level of Robustness selected for this PP.

Section 2 describes, in detail, the WLAN client TOE (i.e., the TOE for this PP) and the IT environment upon which the TOE depends.

Section 3 describes the TOE security environment. This includes

- Secure-use assumptions that describe the presumptive conditions for secure use of the TOE in the a basic robustness environment
- Threats that are to be addressed either completely or partially by the technical countermeasures implemented in the WLAN client.
- Organizational policies that levy further requirements on the TOE.

In addition this section also identifies those threats and policies that are defined as part of the basic robustness environment that the WLAN client does not address

Section 4 defines the security objectives for the WLAN client in a basic robustness environment.

Section 5 contains the functional and assurance requirements derived from the CC, Parts 2 and 3, respectively that must be satisfied by the WLAN client. This section also identifies requirements that are levied on the TOE IT environment.

Section 6 provides a rationale to demonstrate that the information technology security objectives for the TOE and its IT environment satisfy the identified policies and threats. The section then provides rationale to show that the set of requirements are sufficient to meet each objective, and that each security objective is addressed by one or more component requirements. Therefore, the two aforementioned subsections provide arguments that the security objectives and security requirements are both necessary and sufficient, respectively and collectively, to meet the needs dictated by the policies and threats. Section 6 also provides arguments that address any unsatisfied dependencies.

Section 7, Identifies references to noteworthy background and/or supporting materials.

Appendix A is an acronym list that defines frequently used acronyms.

1. Introduction

This Protection Profile (PP) supports future Department of Defense (DoD) procurements of commercial off-the-shelf (COTS) wireless local area network (WLAN) clients that will be used in basic robustness environments. This PP details the policies, assumptions, threats, security objectives, security functional requirements, and security assurance requirements for the WLAN client and its supporting environment. In the case of this PP, the TOE supporting environment is significant. The PP has been written with the assumption that the TOE is a wireless network interface card and any supporting software. However, this assumption is not meant to exclude other instantiations of the TOE, which meet the security requirements stated in this PP. In addition, it is assumed that the TOE is a relatively simple device that is a component of a larger system. As such, the TOE must rely heavily on the TOE IT environment for some protection.

This PP has two primary audiences: Information System Security Engineers (ISSE) and COTS WLAN client product vendors. The ISSE may use this PP to help in designing and assessing installations in which COTS WLAN clients are part of the information system. WLAN client product vendors will use the PP to learn the DoD security requirements for new COTS WLANs being procured.

1.1 Identification

Title:	US Government Protection Profile Wireless Local Area Network (WLAN) Client for Basic Robustness Environments
Version:	1.1
Sponsor:	National Security Agency (NSA)
CC Version:	This PP claims conformance to Common Criteria for Information Technology Security Evaluation (CC) Version 3.1, Part 2 extended and Part 3 conformant to include applicable interpretations.
Evaluation Level:	Evaluation Assurance Level (EAL) 2 augmented with, ALC_FLR.2 (Flaw Remediation).).
Keywords:	radio, basic assurance, wireless, network, wireless local area network, wireless LAN, WLAN, LAN

1.2 TOE Overview

This PP specifies the DoD's information security needs for a Basic Robustness WLAN Client. It is expected that the wireless client will be a component in a larger system (for example, a wireless card installed in a laptop computer). This PP requires privacy and integrity of communications over the WLAN using commercially available cryptographic algorithms. Security administration is the responsibility of the user of each component (i.e., client). The assurance requirements specified in the PP are EAL 2 augmented with Flaw Remediation.

This PP addresses the security requirements for a TOE that provides communication between the wireless user and the wired network and its resources. The security features of the TOE include administration, encryption, and audit. The WLAN client is intended to interface with a WLAN access point or access system in the IT environment. The IT environment is expect to provide for the capability of auditing, management, identification and authentication, and protection functions as defined in Section 5.3.

STs that claim conformance to this PP shall meet a minimum standard of demonstrable-PP conformance as defined in section D3 of part 1.

1.3 TOE Environment Defining Factors

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

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

Note that there are an infinite number of combinations of entity authorization and value of resources; this conceptually "makes sense" because there are an infinite number of potential environments, depending on how the resources are valued by the organization, and the variety of authorizations the organization defines for the associated entities. In the next two sections, these two environmental factors will be related to the robustness required for selection of an appropriate TOE.

1.3.1 Value of Resources

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

1.3.2 Authorization of Entities

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

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

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

1.3.3 Selection of Appropriate Robustness Levels

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

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

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

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

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

The preceding examples demonstrated that it is possible for radically different combinations of entity authorization/resource values to result in a similar likelihood of an attempted compromise. As mentioned earlier, the robustness of a system is an indication of the protection being provided to counter compromise attempts. Therefore, a basic robustness system should be sufficient to counter compromise attempts where the likelihood of an attempted compromise is low. The following chart depicts the "universe" of environments characterized by the two factors discussed in the previous section: on one axis is the authorization defined for the least trustworthy entity, and on the other axis is the highest value of resources associated with the TOE.

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

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

attempted compromise is roughly similar. This is graphically depicted in the following chart.

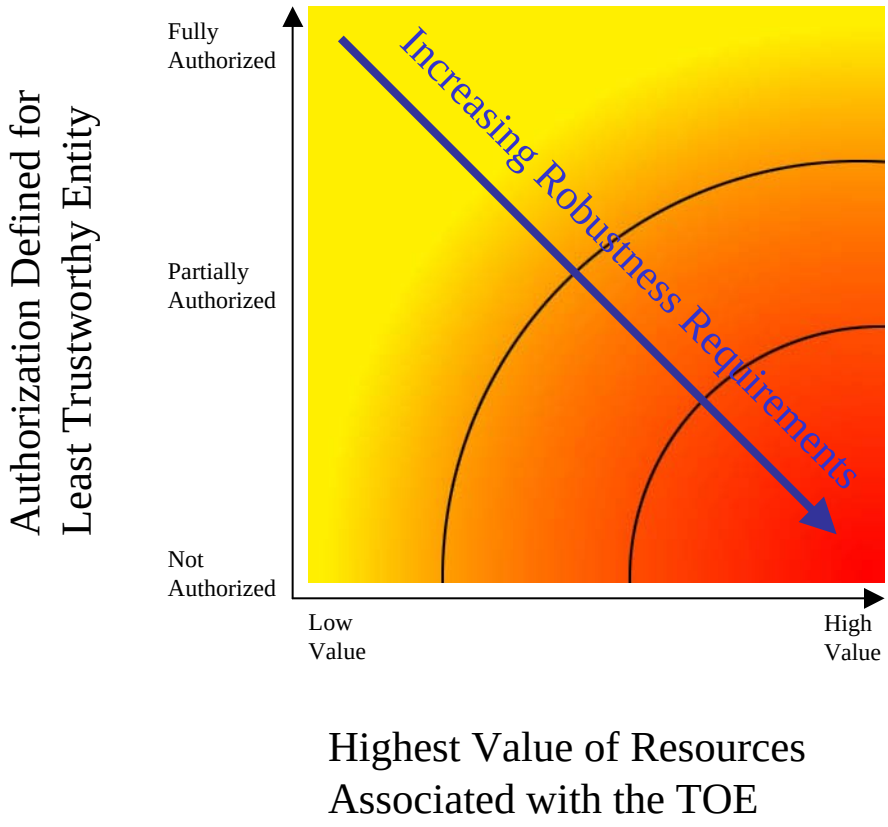

Figure 1: Value of TOE Resources vs. Trust

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

The difficult part of this activity is differentiating the authorization of various entities, as well as determining the relative values of resources; (e.g., what constitutes "low value" data vs. "medium value" data). Because every organization will be different, a rigorous definition is not possible. In section 3 of this PP, the targeted threat level for a basic robustness TOE is characterized. This information is provided to help organizations

using this PP insure that the functional requirements specified by this basic robustness PP are appropriate for their intended application of a compliant TOE.

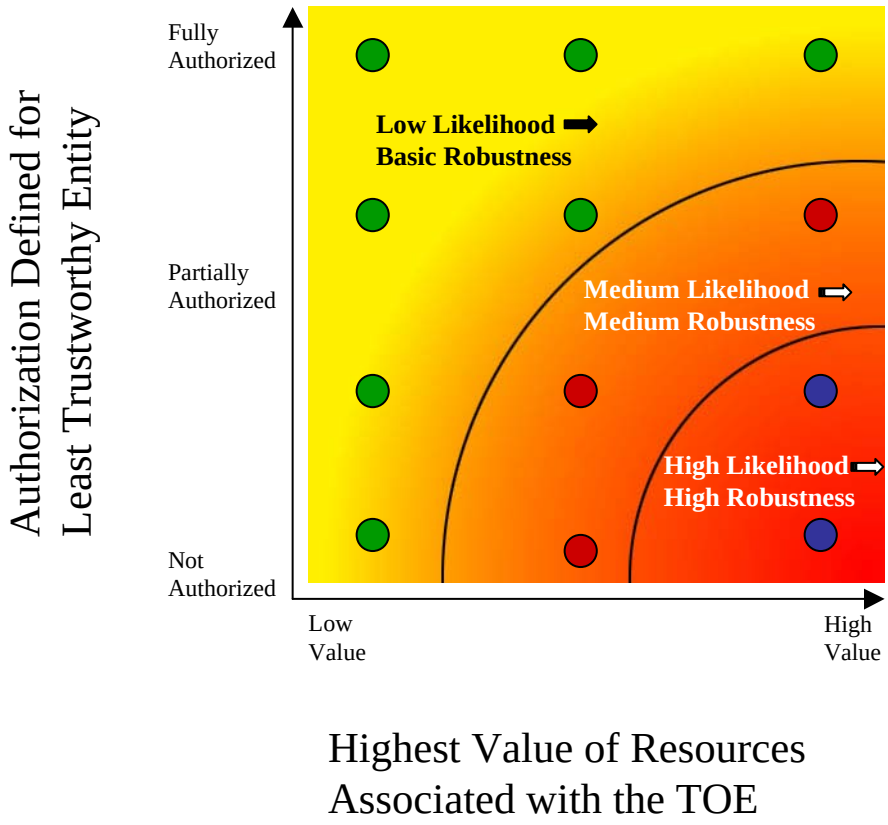

Figure 2: Value of TOE Resources vs. Robustness

1.4 Related Protection Profiles

There are no validated Protection Profiles related to this technology type.

2. TOE Description

A WLAN is an extension, or possibly a replacement, of a traditional wired network. The WLAN client is in most cases installed into the laptop or mobile device. Therefore, it must also be understood that the TOE alone does not provide all of the security functionality that is required in a Basic Robustness Environment. A traditional wireless LAN is set up as in Figure 3. In the typical configuration, the client and access system establish a connection through which all data will traverse to the wired side of the network. As such, it is not intended to provide any direct network services to the users that connect through the access system. The client will rely mainly on the environment in which it resides to perform many of the management duties.

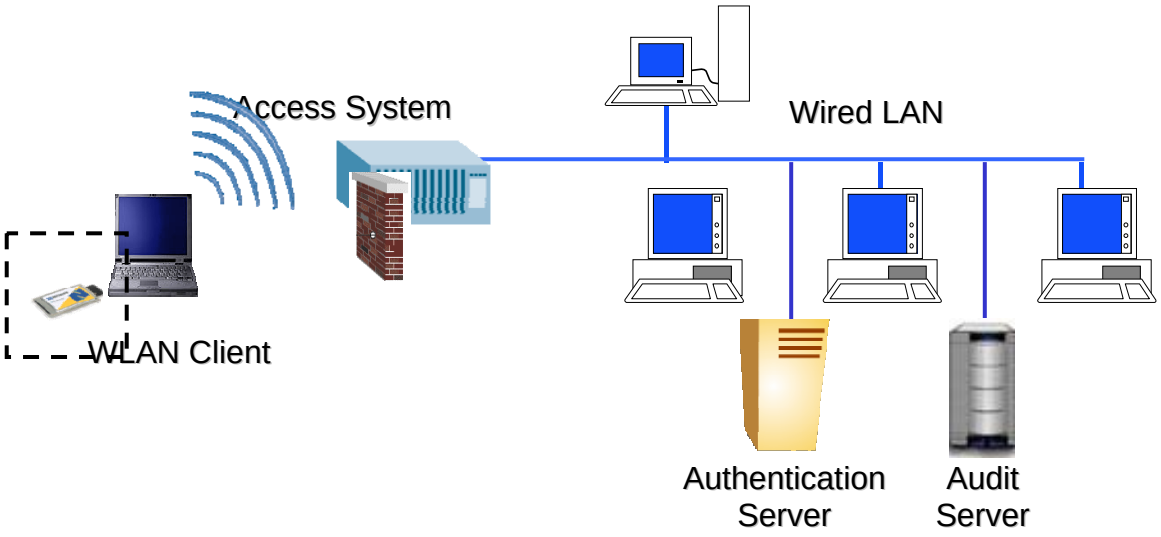

Figure 3: Example of WLAN architecture with the WLAN client

The Target of Evaluation (TOE) is a WLAN client. It is expected that the client will be a component of a larger system (e.g. the WLAN client will be installed on a laptop computer). For the purpose of this PP we will be discussing a typical wired to wireless configuration. However the reader should keep in mind that it does not preclude any other wireless configuration that may exist and meet the requirements in this PP. This PP does not dictate any particular configuration. Instead the PP addresses the security requirements for the client that provides communication between the wireless user and the wired network and its resources.

While this document does not dictate vendor implementations of the functional and assurance requirements defined in Section 5, it is expected that the wireless card, any device drivers necessary to operate the TOE as part of the larger system, and any management software that is used to install, configure or operate the WLAN client be included as part of the TOE in any Security Targets (ST) claiming conformance to this PP. The intent is to ensure vendors/sponsors submit complete products for evaluation rather than restricting the evaluation to specific portions of a product.

The security requirements of the TOE are administration, audit, and encryption. There are additional requirements for the TOE IT environment. Taken together these requirements (for the TOE and its IT environment) mitigate most but not all threats and enforce most but not all policies expected to exist in a basic robustness environment. Table 3 identifies threats that are expected to exist in a Basic Robustness Environment that the WLAN client TOE does not address. Table 5 identifies security policies not addressed by the TOE. Table 8 identifies security objectives for the environment that must be addressed either by assumptions or requirements levied on the TOE IT environment.

2.1 Administration

"Administrator" refers to the roles assigned to the individual(s) responsible for the installation, configuration, and maintenance of the TOE. Since this TOE is part of a larger system, it is expected that those responsible for administration of the TOE IT environment will also be responsible for TOE administration. This PP does not preclude multiple separate administrative roles but requires only a single administrator for the TOE.

2.2 Encryption

This TOE includes requirements for cryptographic modules. Those modules must comply with Federal Information Processing Standard Publication (FIPS PUB) 140-1 or 140-2, which defines security requirements for cryptographic modules. A cryptographic module is that part of a system or application that provides cryptographic services, such as encryption, authentication, or electronic signature generation and verification. Products and systems compliant with this PP are expected to utilize cryptographic modules compliant with this FIPS PUB.

2.3 Audit

This TOE is expected to be a component in a larger computing platform. As such its responsibilities with respect to audit are limited to the generation of audit events. It is expected that the TOE IT environment will provide the mechanisms for audit event storage and retrieval.

2.4 TOE IT Environment

The hardware platform (e.g., handheld PC, notebook computer) in which the WLAN client is installed and the operating system are not required to be included as part of the TOE at basic robustness. However, since the TOE is expected to be part of a larger IT system, it may be necessary for the TOE to rely upon that IT environment to augment the protections offered by the TOE. The specific requirements for the IT environment are identified in section 5.3.

3. TOE Security Environment

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

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

In a basic robustness environment, users are trusted to neither attempt malicious attacks nor by-pass access control measures. Users are also trusted to correctly apply the organization's security policies. The TOE is not expected to protect against sophisticated, technical attack.

Chapter 3 describes the assumptions, threats, and policies that are relevant to both the TOE and the WLAN TOE environment. The first section describes the secure usage assumptions, which are those items that the TOE itself cannot implement or enforce. The next section covers the threats that are expected to exist in a basic robustness environment. The final section discusses the DoD policies relevant to the operation of a WLAN client in a basic robustness environment.

3.1 Secure Usage Assumptions

Assumptions are non-IT items that the TOE itself cannot implement or enforce. Table 1 identifies the assumptions for the WLAN client in the operational environment.

Table 1: TOE Assumptions

Name	Assumption
A.BASIC_ROBUSTNESS_IT_ ENVIRONMENT[4]	The TOE is a Wireless LAN device and is expected to be installed in an IT environment (e.g. PC hardware

[4] This Assumption has been included in the WLAN Client PP in order to ensure that the operational environment in which the TOE is used can address basic robustness threat and policies not addressed by the TOE. It must not be construed as allowing the TOE environment to satisfy TOE functional requirements.

	and O/S) that can appropriately address those threats and policies identified in "Table 3: Basic Robustness Threats NOT Applicable to the TOE" and meets the IT environmental requirements necessary to support the correct operation of the TOE.
A.NO_EVIL	Administrators are non-hostile, appropriately trained and follow all administrator guidance.
A.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the IT environment.

3.2 Threats to Security

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

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

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

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

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

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

The important general points we can make are:

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

The threats listed in Table 2 are general. Exposure of wireless communications in the RF transmission environment introduces unique threats to the WLAN client. With WLANs, an adversary no longer requires physical access to the network in order to exploit a wireless system. The WLAN is susceptible to over-the-air signal intercept, spoofing, and jamming attacks. Given the nature of the basic robustness environment, the threats identified exclude those that would be considered a sophisticated attack (i.e., intentional jamming, traffic analysis).

Table 2: Threats

Threat Name	**Threat Definition**
T.ACCIDENTAL_ADMIN_ ERROR	An administrator may incorrectly install or configure the TOE resulting in ineffective security mechanisms.
T.CRYPTO_COMPROMISE	A user or process may cause key data or executable code associated with the cryptographic functionality to be inappropriately accessed (viewed, modified or deleted), thus compromising the cryptographic mechanisms and the data protected by those mechanisms.
T.POOR_DESIGN	Unintentional errors in requirements specification or design of the TOE may occur, leading to flaws that may be exploited by a casually mischievous user or program.
T.POOR_IMPLEMENTATION	Unintentional errors in implementation of the TOE design may occur, leading to flaws that may be exploited by a casually mischievous user or program.
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.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.TSF_COMPROMISE	A user or process may cause, through an unsophisticated attack, TSF data, or executable code to be inappropriately accessed (viewed, modified, or deleted).

In the case of a WLAN client device, the TOE is a component of a larger system and as such, does not address all of the threats identified as part of a typical basic robustness environment. Table 3 identifies those threats not addressed by the TOE.

Table 3: Basic Robustness Threats NOT Applicable to the TOE

Threat Name	Threat Definition	Rationale for NOT Including this Basic Robustness Threat in the WLAN Client PP
T.AUDIT_COMPROMISE	A user or process may view audit records, cause audit records to be lost or modified, or prevent future audit records from being recorded, thus masking a user's action.	As is noted previously, this TOE is a wireless network interface device, which is installed as part of a larger system. As a component of a larger system, the TOE is responsible for generating audit records in accordance with the audit policy specified by the system administrator. It is expected that these records will be stored outside of the TOE. The TOE IT environment will provide all appropriate mechanisms to protect audit records after they have been generated. Since the TOE (WLAN client) does not contribute to the mitigation of this threat it has not been included in the PP.
T.MASQUERADE	A user or process may masquerade as another entity in order to gain unauthorized access to data or TOE resources.	As is noted previously, this TOE is a wireless network interface card, which is installed as part of a larger system. As a component of a larger system, the TOE (wireless NIC) is not provided with user or process identification information and is not expected to prevent masquerading by an unauthorized user or process. Since the TOE (WLAN client) does not contribute to the mitigation of this threat it has not been included in the PP.

Threat Name	Threat Definition	Rationale for NOT Including this Basic Robustness Threat in the WLAN Client PP
T.UNATTENDED_ SESSION	A user may gain unauthorized access to an unattended session.	The PP authors recognize that this threat, although appropriate for a basic robustness environment, will not be addressed (either fully or partially) by the TOE. The TOE, in this case, is a wireless network interface card, which is installed as part of a larger system. As a component of larger system, the only unattended sessions within the TOE scope of control are network connections. The PP authors believe that this threat is more appropriately mitigated by the operating system in which the WLAN client is installed. The OS is capable of uniformly enforcing a policy for unattended network, serial interface and console sessions. Since the TOE (WLAN client) does not contribute to the mitigation of this threat it has not been included in the PP.
T.UNAUTHORIZED_ ACCESS	A user may gain access to user data for which they are not authorized according to the TOE security policy.	As is noted previously, this TOE is a wireless network interface card, which is installed as part of a larger system. As a component of a larger system, the does not have access to information identifying authorized or unauthorized users. Since the TOE (WLAN client) does not contribute to the mitigation of this threat it has not been included in the PP.
T.UNIDENTIFIED_ ACTIONS	The administrator may not have the ability to notice potential security	As is noted previously, this TOE is a wireless network interface card, which is installed as part of a larger system. As a component

Threat Name	Threat Definition	Rationale for NOT Including this Basic Robustness Threat in the WLAN Client PP
	violations, thus limiting the administrator's ability to identify and take action against a possible security breach.	of a larger system, the TOE is responsible for generating audit records in accordance with the audit policy specified by the system administrator. However the TOE is not expected to provide facilities to either store or review audit records. It is expected that the TOE IT environment will provide facilities to review, sort, select and otherwise manage the audit records. Since the TOE (WLAN client) does not contribute to the mitigation of this threat it has not been included in the PP.

3.3 Organizational Security Policies

An organizational security policy is a set of rules, practices, and procedures imposed by an organization to address its security needs. Table 4 identifies the organizational security policies applicable to the basic robustness WLAN client. PP-compliant TOEs must address the organizational security policies described below.

Table 4: Organizational Security Policies

Policy Name	Policy Definition
P.ACCOUNTABILITY	The authorized users of the TOE shall be held accountable for their actions within the TOE.
P.CRYPTOGRAPHY	Only NIST FIPS validated cryptography (methods and implementations) are acceptable for key management (i.e.; generation, access, distribution, destruction, handling, and storage of keys) and cryptographic services (i.e.; encryption, decryption, signature, hashing, key exchange, and random number generation services).

In the case of a WLAN client device, the TOE is a component of a larger system and as such, does not address all of the policies identified as part of a basic robustness environment. Table 5 identifies those policies.

Table 5: Basic Robustness Policies Not Addressed By the TOE

Policy Name	Policy Definition	Rationale for NOT Including this Basic Robust Policy in the WLAN Client PP
P.ACCESS_BANNER	The TOE shall display an initial banner describing restrictions of use, legal agreements, or any other appropriate information to which users consent by accessing the system.	As is noted previously, this TOE is a wireless network interface card, which is installed as part of a larger system. As such, the TOE IT environment (e.g. operating system) is responsible for the display of appropriate banner information. Since the TOE (WLAN client) does not contribute to the implementation of this policy, it has not been included in the PP.

3.4 Security Function Policies

Several of the functional requirements in section 5.1 reference Security Function Policies (SFPs). SFPs are assigned to a named set of rules enforced by the TOE and described in the Security Functional Requirements. They are not organizational policies. The single SFP applicable to a wireless client is listed in the table below with an explanation that supplies additional information and interpretation.

Table 6 Security Function Policies

Policy Name	Policy Definition
P.WIRELESS CLIENT ENCRYPTION SFP	The users/access system administrators shall specify that the TOE encrypt/decrypt user data as it transits to/from wireless network.

4. Security Objectives for the TOE

Table 7 identifies the security objectives of the TOE. These security objectives reflect the stated intent to counter identified threats and/or comply with any organizational security policies identified.

Table 7: Security Objectives for the TOE

Name	TOE Security Objective	Corresponding Threats or Policies
O.ADMIN_GUIDANCE	The TOE will provide administrators with the necessary information for secure management.	T.ACCIDENTAL_ADMIN _ ERROR
O.AUDIT_GENERATION	The TOE will provide the capability to detect and create records of security-relevant events associated with users.	P.ACCOUNTABILITY
O.CORRECT_ TSF_OPERATION	The TOE will provide the capability to test the TSF to ensure the correct operation of the TSF at a customer's site.	T.POOR_TEST
O.CRYPTOGRAPHY	The TOE shall use NIST FIPS 140-1 or 140-2 validated cryptographic services.	P.CRYPTOGRAPHY, T.CRYPTO_COMPROMIS E
O.MANAGE	The TOE will provide functions and facilities necessary to support the administrators in their management of the security of the TOE.	T.TSF_COMPROMISE
O.RESIDUAL_ INFORMATION	The TOE will ensure that any information contained in a protected resource within its Scope of Control is not released when the resource is reallocated.	T.RESIDUAL_DATA, T.TSF_COMPROMISE, P.CRYPTOGRAPHY, T.CRYPTO_COMPROMIS E

Name	TOE Security Objective	Corresponding Threats or Policies
O.CONFIGURATION_ IDENTIFICATION	The configuration of the TOE is fully identified in a manner that will allow implementation errors to be identified and corrected, with the TOE being redistributed promptly.	T.POOR_DESIGN, T.POOR_IMPLEMENTATION
O.DOCUMENTED_ DESIGN	The design of the TOE is adequately and accurately documented.	T.POOR_DESIGN
O.PARTIAL_ FUNCTIONAL_TESTING	The TOE will undergo some security functional testing that demonstrates the TSF satisfies some of its security functional requirements.	T.POOR_IMPLEMENTATION, T.POOR_TEST
O.VULNERABILITY_ ANALYSIS	The TOE will undergo some vulnerability analysis demonstrate that the design and implementation of the TOE does not contain any obvious flaws.	T.POOR_DESIGN, T.POOR_IMPLEMENTATION, T.POOR_TEST

4.1 Security Objectives for the Environment

This section defines the security objectives that are to be addressed by the IT domain or by non-technical or procedural means. The assumptions identified in Section 3.1 are incorporated as security objectives for the environment. They levy additional requirements on the environment, which are largely satisfied through procedural or administrative measures. Table 8 identifies the security objectives for the environment.

Table 8: Security Objectives for the Environment

Name	TOE Security Objective	Corresponding Assumption, Threat, or Policy

Name	TOE Security Objective	Corresponding Assumption, Threat, or Policy
OE.BASIC_ROBUSTNESS_OS	The TOE is a Wireless LAN card and is expected to be installed in an IT environment (e.g. PC hardware and O/S) that can appropriately address those threats and policies identified in "Table 3: Basic Robustness Threats NOT Applicable to the TOE" and meets the IT environmental requirements necessary to support the correct operation of the TOE.	A.BASIC_ROBUSTNESS_IT_ENVIRONMENT
OE.MANAGE	The TOE IT environment will augment the TOE 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.	T.TSF_COMPROMISE, P.ACCOUNTABILITY, T.ACCIDENTAL_ADMIN_ ERROR
OE.NO_EVIL	Administrators are non-hostile, appropriately trained and follow all administrator guidance.	A.NO_EVIL
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.	A.PHYSICAL
OE.RESIDUAL_INFORMATION	The TOE IT environment will ensure that any information contained in a protected resource within its Scope of Control is not released when the resource is reallocated.	T.TSF_COMPROMISE, T.RESIDUAL_DATA, T.CRYPTO_COMPROMISE
OE.SELF_PROTECTION	The TOE IT environment will maintain a domain for itself and the TOE's own execution that protects them and their resources from external interference, tampering, or unauthorized disclosure through their interfaces.	T.TSF_COMPROMISE, T.CRYPTO_COMPROMISE

Name	TOE Security Objective	Corresponding Assumption, Threat, or Policy
OE.TIME_STAMPS	The TOE IT environment shall provide reliable time stamps and the capability for the administrator to set the time used for these time stamps.	P.ACCOUNTABILITY
OE.TOE_ACCESS	The TOE IT environment will provide mechanisms that control a user's logical access to the TOE.	P.ACCOUNTABILITY

5. IT Security Requirements

This section provides functional and assurance requirements that must be satisfied by a PP-compliant TOE. These requirements consist of functional components from Part 2 of the Common Criteria (CC) and an EAL containing assurance components from Part 3 of the CC.

5.1 Identification of Standards Compliance Methods

For this PP, cryptographic operations and key management functions must meet FIPS 140-1 or 140-2 (Level 1). The designated approval authority of the TOE-user organization will specify the methodology used to show compliance to FIPS 140-1 or 140-2 standards. Authorized certificates used by a PP-compliant TOE must be DoD PKI Class 3 or 4, X.509 certificates.

5.2 TOE Security Functional Requirements

The SFRs for the TOE consist of the following components from Part 2 of the CC, summarized in Table 9. All dependencies among the SFRs are satisfied by the inclusion of the relevant requirement within the TOE security requirements.[5]

Table 9: TOE Security Functional Requirements

Functional Component		Dependencies
FAU_GEN_(EXT).1	Audit Data Generation	FPT_STM.1
FCS_BCM_(EXT).1	Extended: Baseline Cryptographic Module	None
FCS_CKM.1(1)	Cryptographic Symmetric key generation	[FCS_CKM.2 or FCS_COP.1] FCS_CKM.4 FMT_MSA.2
FCS_CKM.1(2)	Cryptographic Asymmetric key generation	[FCS_CKM.2 or FCS_COP.1] FCS_CKM.4 FMT_MSA.2
FCS_CKM.2	Cryptographic key distribution	[FDP_ITC.1 or FCS_CKM.1] FMT_MSA.2
FCS_CKM_(EXT).2	Cryptographic key handling and storage	[FDP_ITC.1 or FCS_CKM.1] FMT_MSA.2

[5]Not all of the dependencies identified are satisfied. Section 6 provides the rationale unsatisfied dependencies.

Functional Component		Dependencies
FCS_CKM.4	Cryptographic key destruction	[FDP_ITC.1 or FCS_CKM.1] FMT_MSA.2
FCS_COP.1(1)	Cryptographic Operation (Data encryption/decryption)	[FDP_ITC.1 or FCS_CKM.1] FCS_CKM.4 FMT_MSA.2
FCS_COP.1(2)	Cryptographic Operation (Digital Signature)	[FDP_ITC.1 or FCS_CKM.1] FCS_CKM.4 FMT_MSA.2
FCS_COP.1(3)	Cryptographic Operation (Hashing)	[FDP_ITC.1 or FCS_CKM.1] FCS_CKM.4 FMT_MSA.2
FCS_COP.1(4)	Cryptographic Operation (Key agreement)	[FDP_ITC.1 or FCS_CKM.1] FCS_CKM.4 FMT_MSA.2
FCS_COP_(EXT).1	Extended: Random Number Generation	[FDP_ITC.1or FCS_CKM.1] FCS_CKM.4 FMT_MSA.2
FDP_IFC.1	Subset information flow control (Wireless Client Encryption Policy)	FDP_IFF.1
FDP_IFF.1	Simple Security Attributes (Wireless Client Policy)	FDP_IFC.1 FMT_MSA.3
FDP_RIP.1	Subset Residual Information Protection	None
FMT_MSA.2	Secure Security Attributes	ADV_SPM.1 FDP_IFC.1 FMT_MSA.1 FMT_SMR.1
FMT_MSA.3	Static Attribute Initialization	FMT_MSA.1 FMT_SMR.1
FMT_SMF.1(1)	Specification of Management Functions (Cryptographic Function)	None
FMT_SMF.1(2)	Specification of Management Functions (Audit Record Generation)	None
FMT_SMF.1(3)	Management of TSF data (Cryptographic Key Data)	None
FPT_TST_(EXT).1	TSF Testing	None

Functional Component		Dependencies
FPT_TST.1	TSF Testing of Cryptographic Modules	None
FPT_TST.2	TSF Testing of Cryptographic Key Generation	None

5.2.1 FAU_GEN_(EXT).1 Extended: Audit Data Generation

FAU_GEN_(EXT).1.1 The TSF shall be able to generate an audit record of the following auditable events:

a) All auditable events listed in Table 10;

Table 10 Auditable Events

Requirement	Auditable Events	Additional Audit Record Contents
FCS_CKM_(EXT).2	Error(s) detected during cryptographic key transfer	None
FCS_CKM.4	Destruction of a cryptographic key	None
FDP_IFC.1	Dropping a packet that fails to satisfy the Wireless Client Encryption Policy	MAC address of source and destination devices
FMT_SMF.1(1)	Changing the TOE encryption algorithm including the selection not to encrypt communications	Encryption algorithm selected (or none)
FMT_SMF.1(3)	Changes to the cryptographic key data	None – the TOE **SHALL NOT** record cryptographic keys in the audit log.
FPT_TST_(EXT).1	Execution of the self test	Success or Failure of test
FPT_TST.1	Execution of the self test	Success or Failure of test
FPT_TST.2	Execution of the self test	Success or Failure of test

FAU_GEN_(EXT).1.2 The TSF shall record within each audit record at least the following information:

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

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [information specified in column three of Table 10 Auditable Events].

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

Application Note: This requirement has been generated as an extended requirement in order to remove the statement that requires the TOE generate audit events that correlate with the startup and shutdown of the audit function. This is not practical for WLAN client devices.

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.

5.2.2 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.

5.2.2.1 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:***

Entirely in hardware shall have a minimum overall rating of FIPS PUB 140-2, Level 3,

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.

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.

5.2.3 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.

5.2.3.1 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)).

5.2.3.2 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:***

a FIPS-Approved Random Number Generator as specified in FCS_COP_(EXT).1, and/or

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.

5.2.3.3 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:***

Manual (Physical) Method, and/or

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.

5.2.3.4 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.

5.2.3.5 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.

5.2.4 Cryptographic Operation (FCS_COP)

5.2.4.1 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*]**.

5.2.4.2 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:***

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

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

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.

5.2.4.3 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.

5.2.4.4 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."

5.2.4.5 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.

5.2.5 FDP_IFC.1 Subset information flow control (Wireless Client Encryption Policy)

FDP_IFC.1.1 The TSF shall enforce the [Wireless Client Encryption Policy] on [subjects: client, access point/system; information: network packets; operations: receive packet and transmit packet].

5.2.6 FDP_IFF.1 Simple Security Attributes (Wireless Client Encryption Policy)

FDP_IFF.1.1 The TSF shall enforce the [Wireless Client Encryption Policy] based on the following types of subject and information security attributes: [subjects: client, access point/system; information: encryption/decryption flag, direction of travel at the network interface]

FDP_IFF.1.2 The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: [

- If the encryption/decryption flag does NOT indicate that the TOE should perform encryption then all packets may pass without modification.
- If the direction of travel is from the operating system to the network interface and the encryption/decryption flag indicates the TOE should perform encryption, then the TOE must encrypt user data via FCS_COP_(EXT).2.1 and if successful transmit the packet via the wireless interface.
- The direction of travel is from the network interface to the operating system and the encryption/decryption flag indicates the TOE should perform encryption then the TOE must decrypt user data via FCS_COP_(EXT).2.1 and if successful pass that information to the operating system.

- [ST AUTHOR - selection: [ST AUTHOR - assignment: for each operation, the security attribute-based relationship that must hold between subject and information security attributes], “no additional information flow Specified Access Point/System Policy Rules”]].

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

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

FDP_IFF.1.5 The TSF shall explicitly authorize an information flow based on the following rules: [ST AUTHOR - selection: [ST AUTHOR - assignment: rules, based on security attributes, that explicitly authorize information flows], "no explicit authorization rules"]

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

Application Note: The encryption/decryption flag identifies a management setting on the TOE.

Application Note:The Wireless Client Encryption Policy implements the encryption policy P.WIRELESS ENCRYPTION SFP described in the TOE environment section of this PP. It is important to note that although the P.WIRELESS ENCRYPTION SFP is clear (the TOE shall encrypt/decrypt wireless traffic when the administrator has required it), the implementation of that policy requires one to consider the direction data is flowing through TOE.

5.2.7 FDP_RIP.1 Subset Residual Information Protection

FDP_RIP.1.1 The TSF shall be ensure that any previous information content of a resource is made unavailable upon the [ST AUTHOR - selection: allocation of the resource to, deallocation of the resource from] the following objects [network packet objects].

Application Note: This requirement ensures that the TOE does not allow data from a previously transmitted packet to be inserted into unused areas or padding in the current packet. Similarly, the TOE must ensure that the contents of previously transmitted packet be cleared from shared memory or other mechanisms (within the TSC) used to transfer packet data between the TOE and the computer in which the TOE is installed.

5.2.8 FMT_MSA.2 Secure security attributes

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

5.2.9 FMT_MSA.3 Static Attribute Initialization

FMT_MSA.3.1 The TSF shall enforce the [Wireless Client Encryption Policy] to provide *restrictive* default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 The TSF shall allow the [Administrator] to specify alternative initial values to override the default values when an object or information is created.

5.2.10 FMT_SMF.1(1) Specification of Management Functions[6] (Cryptographic Function)

FMT_SMF.1.1(1) The TSF shall be capable of performing the following security management functions: [set the encryption/decryption of network packets (via FCS_COP_(EXT).2) in conformance with the Wireless Client Policy].

Application Note: This requirement ensures that those responsible for TOE administration are able to select an encryption algorithm identified in FCS_COP_(EXT).2 or no encryption for encrypting/decrypting data transmitted by the WLAN card.

5.2.11 FMT_SMF.1(2) Specification of Management Functions (TOE Audit Record Generation)

FMT_SMF.1.1(2) The TSF shall be capable of performing the following security management functions: [enable or disable Security Audit (FAU_GEN_(EXT).1)].

Application Note: This requirement ensures that those responsible for TOE administration are able to start or stop the TOE generation of audit records

5.2.12 FMT_SMF.1(3) Specification of Management Functions (Cryptographic Key Data)

FMT_SMF.1.1(3) The TSF shall be capable of performing the following security management functions: [set, modify, and delete the cryptographic keys and key data in support of the Wireless Client Policy and enable/disable verification of cryptographic key testing].

[6] The FMT_SMF (Specification of Management Functions) family is documented in CCIMB interpretation 65.

Application Note: The intent of this requirement is to provide the ability to configure the TOE's cryptographic key(s). Configuring the key data may include: setting key lifetimes, setting key length, etc.

5.2.13 Explicit: 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 .

5.2.14 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:**[i]

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**.[ii]

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 .

5.2.15 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 .

5.3 TOE IT Environment Security Functional Requirements

Table 11 Security Functional Requirements for the TOE IT Environment

Functional Component		Dependencies[7]
FAU_GEN.2	User identity association	FAU_GEN.1 FIA_UID.1
FAU_SAA.1	Potential violation analysis	FAU_GEN.1
FAU_SAR.1	Audit Review	FAU_GEN.1
FAU_SAR.2	Restricted Audit Review	FAU_SAR.1
FAU_SAR.3	Selectable audit review	FAU_SAR.1
FAU_SEL.	Selective audit	FAU_GEN.1 FMT_MTD.1
FAU_STG.1	Protected audit trail storage	FAU_GEN.1
FAU_STG.3	Action in case of possible audit data loss	FAU_STG.1
FIA_USB.1	User-subject Binding	FIA_ATD.1
FMT_MOF.1	Management of Security Functions Behavior	FMT_SMR.1
FMT_MTD.1	Management of TSF Data (Time TSF Data)	FMT_SMR.1
FMT_SMR.1	Security Roles	FIA_UID.1
FDP_RIP.1	Subset Residual Information Protection	None
FPT_STM.1	Reliable Time Stamps	None

Application Note: This protection profile requires that the TOE IT environment provide significant functionality. It is also acceptable for an ST claiming compliance with this PP to satisfy some or all of the requirements levied on the IT environment by including the same requirements as part of the TOE.

5.3.1 FAU_GEN.2 User identity association

FAU_GEN.2.1 The **TOE IT environment** shall be able to associate each auditable event with the identity of the user that caused the event.

5.3.2 FAU_SAA.1 Potential violation analysis

FAU_SAA.1.1 The **TOE IT environment** 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

[7] The purpose of requirements on the IT environment is to supplement the TOE and to ensure that the TOE and the IT environment together satisfy all security objectives. In order to limit the scope of the IT environment only those IT environmental requirements that directly contribute to the satisfaction of objectives have been included in this PP. Requirements for the IT environment that are necessary simply to satisfy management guidance, audit guidance or dependency chains have not been included in this PP.

FAU_SAA.1.2 The **TOE IT environment** shall enforce the following rules for monitoring audited events:

a) Accumulation **of a single auditable event** or combination of [auditable events in Table 10] known to indicate a potential security violation;

b) *no additional rules*

5.3.3 FAU_SAR.1 Audit review

FAU_SAR.1.1 The **TOE IT environment** shall provide **only** the [Administrator] with the capability to read [all audit data] from the audit records.

FAU_SAR.1.2 Refinement: The TOE IT environment shall provide the audit records in a manner suitable for the **Administrator** to interpret the information.

Application Note: This requirement ensures that the TOE IT environment provides the administrator with functionality necessary for the administrator to review the audit records generated by the TOE.

5.3.4 FAU_SAR.2 Restricted audit review

FAU_SAR.2.1 The **TOE IT environment** shall prohibit all users read access to the audit records, except those users that have been granted explicit read-access.

Application Note: This requirement ensures that access to audit records generated by the TOE is limited to those authorized to view the information.

5.3.5 FAU_SAR.3 Selectable audit review

FAU_SAR.3.1 The **TOE IT environment** shall provide the ability to perform *searches, sorting, ordering* of audit data based on [criteria with logical relations].

5.3.6 FAU_SEL.1 Selective audit

FAU_SEL.1.1 **The** TOE IT environment **shall be able to include or exclude auditable events from the set of audited events based on the following attributes:**

a.) [selection: *subject, identity, host identity*] .

b.) [ST AUTHOR assignment: additional selectable audit attributes].

5.3.7 FAU_STG.1 Protected audit trail storage

FAU_STG.1.1 The **TOE IT environment** shall protect the stored audit records from unauthorized deletion.

FAU_STG.1.2 The **TOE IT environment** shall be able to *prevent* unauthorized modifications to the audit records in the audit trail.

5.3.8 FAU_STG.3 Action in case of possible audit data loss

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

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

5.3.9 FIA_USB.1 User-subject binding

FIA_USB.1.1 The **TOE IT environment** shall associate the following user security attributes with subjects acting on the behalf of that user: [authentication credentials].

FIA_USB.1.2 The **TOE IT environment** shall enforce the following rules on the initial association of user security attributes with subjects acting on the behalf of users: [ST AUTHOR - assignment: *rules for the initial association of attributes*].

FIA_USB.1.3 The **TOE IT environment** shall enforce the following rules governing changes to the user security attributes associated with subjects acting on behalf of users: [ST AUTHOR - assignment: *rules for the changing of attributes*].

5.3.10 FMT_MTD.1 Management of TSF Data (Time TSF Data)

FMT_MTD.1.1 The **TOE IT environment** shall restrict the ability to *set* the [time and date used to form the time stamps in FPT_STM.1] to [the Administrator].

Application Note: The TOE IT environment must provide an interface for the Administrator to set the time and date.

5.3.11 FDP_RIP.1 Subset Residual Information Protection

FDP_RIP.1.1 The **TOE IT environment** shall ensure that any previous information content of a resource is made unavailable upon the *allocation of the resource to* the following objects [network packet objects].

Application Note: This requirement ensures that the TOE IT Environment does not allow data from a previously transmitted packet to be inserted into unused areas or padding in the current packet. Since operations on requirements for the IT environment must be completed, the selection "allocation of the resource to" has been made because it encompasses the two options (e.g. a system that make the information contents of resource unavailable when the resource is freed can also claim to meet the requirement that the content of the resource be freed prior to reallocation).

5.3.12 FMT_SMR.1 Security Roles

FMT_SMR.1.1 The **TOE IT environment** shall maintain the role [Administrator].

FMT_SMR.1.2 The **TOE IT environment** shall be able to associate users with roles.

Application Note: The TOE IT environment provides support for the administrative role that is used to administer the TOE. In some environments, the administrative role will be fulfilled by the end user (e.g. a laptop computer). However, other environments (e.g. a multi-user system), the administrative role will be provided by someone other than the end user.

5.3.13 FPT_STM.1 Reliable Time Stamps

FPT_STM.1.1 The **TOE IT environment** shall be able to provide reliable time **and date** stamps for **the TOE and** its own use.

Application Note: The TOE IT environment must provides time stamps that are used by the TOE.

5.4 TOE Security Assurance Requirements

The TOE security assurance requirements, summarized in Table 12, identify the management and evaluative activities required to address the threats and policies identified in section 3 of this protection profile. Section 6.5 provides a justification for the chosen security assurance requirements and the selected EAL 2 assurance level.

Table 12: TOE Assurance Requirements

Assurance Class	ASSURANCE COMPONENTS	ASSURANCE COMPONENTS DESCRIPTION
DEVELOPMENT	ADV_ARC.1	Architectural Design with domain separation and non-bypassability
	ADV_FSP.2	Security-enforcing Functional Specification
	ADV_TDS.1	Basic design
GUIDANCE DOCUMENTS	AGD_OPE.1	Operational user guidance
	AGD_PRE.1	Preparative User guidance
LIFE CYCLE SUPPORT	ALC_CMC.2	Use of a CM system
	ALC_CMS.2	Parts of the TOE CM coverage
	ALC_DEL.1	Delivery procedures
	ALC_FLR.2	**Flaw Reporting Procedures**
TESTS	ATE_COV.1	Evidence of coverage
	ATE_FUN.1	Functional testing
	ATE_IND.2	Independent testing - conformance
VULNERABILITY ASSESSMENT	AVA_VAN.2	Vulnerability analysis

5.4.1 Class ADV: Development

5.4.1.1 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.

5.4.1.2 ADV_FSP.2 Security-enforcing functional specification

Dependencies: ADV_TDS.1 Basic design

Developer action elements:

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

ADV_FSP.2.2D The developer shall provide a tracing from the functional specification to the SFRs.

Content and presentation elements:

ADV_FSP.2.1C The functional specification shall completely represent the TSF.

ADV_FSP.2.2C The functional specification shall describe the purpose and method of use for all TSFI.

ADV_FSP.2.3C The functional specification shall identify and describe all parameters associated with each TSFI.

ADV_FSP.2.4C For each SFR-enforcing TSFI, the functional specification shall describe the SFR-enforcing actions associated with the TSFI.

ADV_FSP.2.5C For SFR-enforcing TSFIs, the functional specification shall describe direct error messages resulting from processing associated with the SFR-enforcing actions.

ADV_FSP.2.6C The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification.

Evaluator action elements:

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

ADV_FSP.2.2E The evaluator *shall determine* that the functional specification is an accurate and complete instantiation of the SFRs.

5.4.1.3 ADV_TDS.1 Basic design

Dependencies: ADV_FSP.2 Security-enforcing functional specification

Developer action elements:

ADV_TDS.1.1D The developer shall provide the design of the TOE.

ADV_TDS.1.2D 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:

ADV_TDS.1.1C The design shall describe the structure of the TOE in terms of subsystems.

ADV_TDS.1.2C The design shall identify all subsystems of the TSF.

ADV_TDS.1.3C The design shall describe the behavior of each SFR-supporting or SFR-non-interfering TSF subsystem in sufficient detail to determine that it is not SFR-enforcing.

ADV_TDS.1.4C The design shall summarize the SFR-enforcing behavior of the SFR-enforcing subsystems.

ADV_TDS.1.5C The design shall provide a description of the interactions among SFR-enforcing subsystems of the TSF, and between the SFR-enforcing subsystems of the TSF and other subsystems of the TSF.

ADV_TDS.1.6C The mapping shall demonstrate that all behavior described in the TOE design is mapped to the TSFIs that invoke it.

Evaluator action elements:

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

ADV_TDS.1.2E The evaluator *shall determine* that the design is an accurate and complete instantiation of all security functional requirements.

5.4.2 Class AGD: Guidance documents

5.4.2.1 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.

5.4.2.2 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.

5.4.3 Class ALC: Life-cycle support

5.4.3.1 ALC_CMC.2 Use of a CM system

Dependencies: ALC_CMS.1 TOE CM coverage

Developer action elements:

ALC_CMC.2.1D The developer shall provide the TOE and a reference for the TOE.

ALC_CMC.2.2D The developer shall provide the CM documentation.

ALC_CMC.2.3D The developer shall use a CM system.

Content and presentation elements:

ALC_CMC.2.1C The TOE shall be labeled with its unique reference.

ALC_CMC.2.2C The CM documentation shall describe the method used to uniquely identify the configuration items.

ALC_CMC.2.3C The CM system shall uniquely identify all configuration items.

Evaluator action elements:

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

5.4.3.2 ALC_CMS.2 Parts of the TOE CM coverage

Dependencies: No dependencies.

Developer action elements:

ALC_CMS.2.1D The developer shall provide a configuration list for the TOE.

Content and presentation elements:

ALC_CMS.2.1C The configuration list shall include the following: the TOE itself; the evaluation evidence required by the SARs; and the parts that comprise the TOE.

ALC_CMS.2.2C The configuration list shall uniquely identify the configuration items.

ALC_CMS.2.3C For each TSF relevant configuration item, the configuration list shall indicate the developer of the item.

Evaluator action elements:

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

5.4.3.3 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.

5.4.3.4 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.

5.4.4 Class ATE: Tests

5.4.4.1 ATE_COV.1 Evidence of coverage

Dependencies: ADV_FSP.2 Security-enforcing functional specification
ATE_FUN.1 Functional testing

Developer action elements:

ATE_COV.1.1D The developer shall provide evidence of the test coverage.

Content and presentation elements:

ATE_COV.1.1C The evidence of the test coverage shall show the correspondence between the tests in the test documentation and the TSFIs in the functional specification.

Evaluator action elements:

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

5.4.4.2 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.

5.4.4.3 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.

5.4.5 Class AVA: Vulnerability assessment

5.4.5.1 AVA_VAN.2 Vulnerability analysis

Dependencies: ADV_ARC.1 Security architecture description
ADV_FSP.1 Basic functional specification
ADV_TDS.1 Basic design
AGD_OPE.1 Operational user guidance
AGD_PRE.1 Preparative procedures

Developer action elements:

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

Content and presentation elements:

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

Evaluator action elements:

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

AVA_VAN.2.2E The evaluator *shall perform* a search of public domain sources to identify potential vulnerabilities in the TOE.

AVA_VAN.2.3E The evaluator *shall perform* an independent vulnerability analysis of the TOE using the guidance documentation, functional specification, TOE design and security architecture description to identify potential vulnerabilities in the TOE.

AVA_VAN.2.4E 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 Basic attack potential.

Application Note: The TOE version used as the basis for testing should include a reference to the specific signature set in place when this activity is conducted.

6. Rationale

This section describes the rationale for the Security Objectives and Security Functional Requirements as defined in Section 4 and Section 5, respectively. Additionally, this section describes the rationale for not satisfying all of the dependencies. Table 13 illustrates the mapping from Security Objectives to Threats and Policies.

6.1 Rationale for TOE Security Objectives

Table 13: Security Objectives to Threats and Policies Mappings

Threat/Policy	Objectives Addressing the Threat	Rationale
T.ACCIDENTAL_ADMIN_ERROR An administrator may incorrectly install or configure the TOE resulting in ineffective security mechanisms.	O.ADMIN_GUIDANCE The TOE will provide administrators with the necessary information for secure management.	O.ADMIN_GUIDANCE helps to mitigate this threat by ensuring the TOE administrators have guidance that instructs them how to administer the TOE in a secure manner. Having this guidance helps to reduce the mistakes that an administrator might make that could cause the TOE to be configured in a way that is insecure.
T.CRYPTO_COMPROMISE A user or process may cause key data or executable code associated with the cryptographic functionality to be inappropriately accessed (viewed, modified or deleted), thus compromising the cryptographic mechanisms and the date protected by those mechanisms.	O.RESIDUAL_INFORMATION The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated. OE.RESIDUAL_INFORMATION The TOE IT environment will ensure that any information contained in a protected resource within its Scope of Control is not released when the resource is reallocated. O.CRYPTOGRAPHY The TOE shall use NIST FIPS 140-1 or 140-2 validated cryptographic services. OE.SELF_PROTECTION The TOE IT environment will maintain a domain for itself and the TOE's own execution that protects them and their resources from external interference, tampering, or unauthorized disclosure through its their interfaces.	O.RESIDUAL_INFORMATION and OE.RESIDUAL_INFORMATION contribute the mitigation of this threat by ensuring that neither the TOE nor the TOE IT environment will insert critical data (including data related to encryption) and executable code as padding in network packet objects. O.CRYPTOGRAPHY ensures that FIPS 140-1 or 140-2 procedures are followed when cryptographic keys are handled and destroyed. OE.SELF_PROTECTION ensures that the TOE IT environment will protect the TOE and itself from users.

Threat/Policy	Objectives Addressing the Threat	Rationale
T.POOR_DESIGN Unintentional errors in requirements specification or design of the TOE may occur, leading to flaws that may be exploited by a casually mischievous user or program.	O.DOCUMENTED_ DESIGN The design of the TOE is adequately and accurately documented. O.CONFIGURATION_ IDENTIFICATION The configuration of the TOE is fully identified in a manner that will allow implementation errors to be identified and corrected, with the TOE being redistributed promptly. O.VULNERABILITY_ ANALYSIS The TOE will undergo some vulnerability analysis to demonstrate that the design and implementation of the TOE does not contain any obvious flaws.	O.DOCUMENTED_DESIGN counters this threat, to a degree, by requiring that the TOE be developed using sound engineering principles. The use of a high level design and the functional specification ensure that those responsible for TOE development understand the overall design of the TOE. This in turn decreases the likelihood of design flaws and increases the chance that accidental design errors will be discovered. ADV_RCR.1, which supports O.DOCUMENTED_DESIGN, ensures that the TOE design is consistent across the High Level Design and the Functional Specification. O.CONFIGURATION_IDENTIFICATIO N plays a role in countering this threat by requiring the developer to provide control of the changes made to the TOE's design. O.VULNERABILITY_ANALYSIS ensure that the TOE has been analyzed for obvious vulnerabilities and that any vulnerabilities found have been removed or otherwise mitigated. This includes analysis of any probabilistic or permutational mechanisms incorporated into a TOE claiming conformance to this PP.

Threat/Policy	Objectives Addressing the Threat	Rationale
T.POOR_IMPLEMENTATION Unintentional errors in implementation of the TOE design may occur, leading to flaws that may be exploited by a casually mischievous user or program.	O.CONFIGURATION_ IDENTIFICATION The configuration of the TOE is fully identified in a manner that will allow implementation errors to be identified and corrected, with the TOE being redistributed promptly. O.PARTIAL_FUNCTIONAL_TESTING The TOE will undergo some security functional testing that demonstrates the TSF satisfies some of its security functional requirements. O.VULNERABILITY_ ANALYSIS The TOE will undergo some vulnerability analysis demonstrate the design and implementation of the TOE does not contain any obvious flaws.	O.CONFIGURATION_IDENTIFICATIO N plays a role in countering this threat by requiring the developer to provide control of the changes made to the TOE's design. This ensures that changes to the TOE are performed in a structured manner and tracked. O.PARTIAL_FUNCTIONAL_TESTING ensures that the developers testing of the TOE is sufficient to address all TOE Security Functional requirements. This objective also contributes to removing this threat by ensuring that the security relevant portions of the TOE have been tested against the security functional requirements. O.VULNERABILITY_ANALYSIS ensures that the TOE has been analyzed for obvious vulnerabilities and that the TOE is resistant casually mischievous users. This includes analysis of any probabilistic or permutational mechanisms incorporated into a TOE claiming conformance to this PP.

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

Threat/Policy	Objectives Addressing the Threat	Rationale
T.RESIDUAL_DATA A user or process may gain unauthorized access to data through reallocation of TOE resources from one user or process to another.	O.RESIDUAL_ INFORMATION The TOE will ensure that any information contained in a protected resource within its Scope of Control is not released when the resource is reallocated. OE.RESIDUAL_INFORMATION The TOE IT environment will ensure that any information contained in a protected resource within its Scope of Control is not released when the resource is reallocated. O.CRYPTOGRAPHY The TOE shall use NIST FIPS 140-1 or 140-2 validated cryptographic services.	O.RESIDUAL_INFORMATION The TOE contributes to the mitigation of this threat by ensuring that network packet objects are cleared prior to use. When considering residual information, the resources of interest within the TOE scope of control are network packets. O.CRYPTOGRAPHY Protection is also provided for cryptographic objects via. FIPS 140 compliance (via FCS_CKM_(EXT).2 and FCS_CKM.4) ensures that objects used to store cryptographic keys are overwritten when those keys are no longer needed. OE.RESIDUAL_INFORMATION contributes to the mitigation of this threat by ensuring that neither the TOE nor the TOE IT environment will insert critical data (including data related to encryption) and executable code as padding in network packet objects.

Threat/Policy	Objectives Addressing the Threat	Rationale
T.TSF_COMPROMISE A user or process may cause, through an unsophisticated attack, TSF data, or executable code to be inappropriately accessed (viewed, modified, or deleted).	O.MANAGE The TOE will provide functions and facilities necessary to support the administrators in their management of the security of the TOE. OE.MANAGE The TOE IT environment will augment the TOE functions and facilities necessary to support the administrators in their management of the security of the TOE, and restrict these functions and facilities from unauthorized use. O.RESIDUAL_ INFORMATION The TOE will ensure that any information contained in a protected resource within its Scope of Control is not released when the resource is reallocated. OE.RESIDUAL_INFORMATION The TOE IT environment will ensure that any information contained in a protected resource within its Scope of Control is not released when the resource is reallocated. O.CRYPTOGRAPHY The TOE shall use NIST FIPS 140-1 or 140-2 validated cryptographic services. OE.SELF_PROTECTION The TOE IT environment will maintain a domain for itself and the TOE's own execution that protects them and their resources from external interference, tampering, or unauthorized disclosure through its their interfaces.	O.MANAGE and OE.MANAGE mitigate this threat by restricting access to administrative functions and TSF data to the administrator. O.RESIDUAL_INFORMATION, OE. RESIDUAL_INFORMATION and O.CRYPTOGRAPHY contributes to the mitigation of this threat by ensuring that any residual data is removed from network packet objects and ensuring that cryptographic material is not accessible once it is no longer needed. OE.SELF_PROTECTION requires that the TOE IT environment be able to protect itself and the TOE from tampering and that the security mechanisms in the TOE cannot be bypassed. Without this objective, there could be no assurance that users could not view or modify TSF data or TSF executables.

Threat/Policy	Objectives Addressing the Threat	Rationale
P.ACCOUNTABILITY The authorized users of the TOE shall be held accountable for their actions within the TOE.	O.AUDIT_GENERATION The TOE will provide the capability to detect and create records of security-relevant events associated with users.. O.MANAGE The TOE will provide functions and facilities necessary to support the administrators in their management of the security of the TOE. OE.MANAGE The TOE IT environment will augment the TOE 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. OE.TIME_STAMPS The TOE IT environment shall provide reliable time stamps and the capability for the administrator to set the time used for these time stamps. OE.TOE_ACCESS The TOE IT environment will provide mechanisms that control a user's logical access to the TOE.	O.AUDIT_GENERATION ensures that the TOE is capable of generating records of audit events associated with users. O.MANAGE ensures that the administrator can enable or disable the audit function. OE.MANAGE ensures that the administrator can review the audit event log and restricts access to this information to the administrator. OE.TIME_STAMPS plays a role in supporting this policy by requiring the TOE IT environment provide a reliable time stamp (configured locally by the Administrator or via an external NTP server). The audit mechanism is required to include the current date and time in each audit record. OE.TOE_ACCESS supports this policy by ensuring that the TOE IT environment provides an administrative role and provides a mechanism to identify processes acting on behalf of the administrator.
P.CRYPTOGRAPHY Only NIST FIPS validated cryptography (methods and implementations) are acceptable for key management (i.e.; generation, access, distribution, destruction, handling, and storage of keys) and cryptographic services (i.e.; encryption, decryption, signature, hashing, key exchange, and random number generation services).	O.CRYPTOGRAPHY The TOE shall use NIST FIPS 140-1 or 140-2 validated cryptographic services.	O.CRYPTOGRAPHY satisfies this policy by requiring the TOE to implement NIST FIPS validated cryptographic services. These services will provide confidentiality and integrity protection of TSF data while in transit. O.RESIDUAL_INFORMATION satisfies this policy by ensuring that cryptographic data are cleared according to FIPS 140-1 or 140-2.

6.2 Rationale for TOE Security Requirements

Table 14: Rationale for TOE Security Requirements

Objective	Requirements Addressing the Objective	Rationale
O.ADMIN_GUIDANCE The TOE will provide administrators with the necessary information for secure management.	ALC_DEL.1 AGD_PRE.1 AGD_OPE.1	ALC_DEL.1 ensures that the administrator has the ability to begin their TOE installation with a *clean* (e.g., malicious code has not been inserted once it has left the developer's control) version of the TOE, which is necessary for secure management of the TOE. 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 Preparative User Guidance (AGD_PRE) documentation ensures that once the administrator has followed the installation and configuration guidance the result is a TOE in a secure configuration. The AGD_OPE.1 requirement ensures that the developer provides the administrator with guidance on how to operate the TOE in a secure manner. This includes describing the interfaces used in managing the TOE, and any security parameters that are configurable by the administrator. The documentation also provides a description of how to setup and review the auditing features of the TOE. The AGD_OPE.1 is also intended for non-administrative users. If the TOE provides facilities/interfaces for this type of user, this guidance will describe how to use those interfaces securely. AGD_OPE.1 AND AGD_PRE.1 analysis during evaluation will ensure that the guidance documentation can be followed unambiguously to ensure the TOE is not misconfigured in an insecure state due to confusing guidance.
O.AUDIT_GENERATION The TOE will provide the capability to detect and create records of security-relevant events associated with users.	FAU_GEN_(EXT).1	FAU_GEN_(EXT).1 defines the set of events that the TOE must be capable of recording. This requirement ensures that the Security 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, as well as the additional information that must be recorded for each auditable event.
O.CONFIGURATION_	ALC_CMC.2ALC_CMS.	ALC_CMC.2contributes to this objective by

Objective	Requirements Addressing the Objective	Rationale
IDENTIFICATION The configuration of the TOE is fully identified in a manner that will allow implementation errors to be identified and corrected, with the TOE being redistributed promptly.	2 ALC_FLR.2	requiring the developer have a configuration management plan that describes how changes to the TOE and its evaluation deliverables are managed. ALC_CMS.2 is necessary to define the items that 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, and CM documentation are tracked by the CM system. ALC_FLR.2 plays a role in satisfying 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 discovery by others. The flaw remediation process used by the developer corrects any discovered flaws, performs an analysis to ensure new flaws are not created while fixing the discovered flaws and makes the patch/modified TOE available to users.
O.CORRECT_ TSF_OPERATION The TOE will provide the capability to test the TSF to ensure the correct operation of the TSF at a customer's site.	FPT_TST_(EXT).1 FPT_TST.1(1) FPT_TST.1(2)	FPT_TST_(EXT).1 is 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. The FPT_TST.1(1) for crypto and FPT_TST.1(2) for key generation functional requirement has been included to address the critical nature and specific handling of the cryptographic related TSF data. Since the cryptographic TSF data has specific FIPS PUB requirements associated with them it is important to ensure that any fielded testing on the integrity of these data maintains the same level of scrutiny as specified in the FCS functional requirements.
O.CRYPTOGRAPHY The TOE shall use NIST FIPS 140-1 or 140-2 validated cryptographic services.	FCS_CKM.1(1) FCS_CKM.1(2) FCS_CKM.2 FCS_CKM.4 FCS_CKM_(EXT).2 FCS_BCM_(EXT).1 FCS_COP.1(1) FCS_COP.1(2) FCS_COP.1(3) FCS_COP.1(4) FCS_COP_(EXT).1 FDP_IFC.1 FDP_IFF.1	Baseline cryptographic services are provided in the TOE by FIPS PUB 140-2 compliant modules implemented in hardware, in software, or in hardware/software combinations [FCS_BCM_(EXT).1]. The cryptographic services offered by this baseline capability are augmented and customized in the TOE to support medium robustness environments. These TOE services are based primarily upon functional security requirements in the areas of key management and cryptographic operations. In the area of key management there are functional requirements that address the generation of symmetric keys [FCS_CKM.1 (1)], and the generation of asymmetric keys [FCS_CKM.1 (2)]; methods of manual and automated cryptographic key distribution

Objective	Requirements Addressing the Objective	Rationale
		[FCS_CKM.2]; cryptographic key destruction [FCS_CKM.4]; techniques for cryptographic key validation and packaging [FCS_CKM.1]; and cryptographic key handling and storage [FCS_CKM_(EXT).2]. Specific functional requirements in the area of cryptographic operations address data encryption and decryption [FCS_COP.1 (1)]; cryptographic signatures [FCS_COP.1 (2)]; cryptographic hashing [FCS_COP.1 (3)]; cryptographic key agreement [FCS_COP.1 (4)]; and improved random number generation [FCS_COP_(EXT).1]. FDP_IFC.1 and FDP_IFF.1 identify the policy that the TOE must implement to encrypt/decrypt user data.
O.DOCUMENTED_ DESIGN The design of the TOE is adequately and accurately documented.	ADV_FSP.2 ADV_TDS.1	ADV_FSP.2 requires that the security relevant interfaces to the TSF be completely specified. In this TOE, a complete specification of the network interface is critical in understanding what functionality is presented to untrusted users and how that functionality fits into the enforcement of security policies. Having a complete understanding of what is available at the TSF interface allows one to analyze this functionality in the context of design flaws. ADV_TDS.1 requires that a high-level 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 to provide a general understanding of the TOE and how it functions. ADV_TDS.1 is also used to ensure that the decomposition of the TOE's design are consistent with one another. This is important, since design decisions that are analyzed and made at one level (high level design) that are not correctly or completely realized at a lower level (the functional specification) may lead to a design flaw. This requirement helps in the design analysis to ensure design decisions are realized at across the design. A complete and accurate description of the TOE design is critical to understanding the TOE design. It is this understanding, gained is from the design analysis, which the evaluator relies upon during testing and vulnerability analysis activities.
O.MANAGE The TOE will provide functions and facilities necessary to support the	FMT_MSA.2 FMT_MSA.3 FMT_SMF.1(1)	The FMT requirements are used to satisfy the management objective, as well as other objectives that specify the control of functionality. The

Objective	Requirements Addressing the Objective	Rationale
administrators in their management of the security of the TOE.	FMT_SMF.1(2) FMT_SMF.1(3)	requirements' rationale for this objective focuses on the administrator's capability to perform management functions in order to control the behavior of security functions. FMT_MSA.2 helps to meet the objective by preventing the administrator from erroneously giving an insecure value to a security attribute. FMT_MSA.3 requires that default values used for security attributes are restrictive, and that the Administrator has the ability to override those values. FMT_SMF.1(1) and FMT_SMF.1(3)ensures that the administrator has the ability to control the use of encryption when the TOE is communicating with external systems. FMT_SMF.1(2) provides the administrator with control of the TOE audit record generation mechanism.
O.PARTIAL_FUNCTIONAL_TESTING The TOE will undergo some security functional testing that demonstrates the TSF satisfies some of its security functional requirements.	ATE_COV.1 ATE_FUN.1 ATE_IND.2	In order to satisfy O.FUNCTIONAL_TESTING, the ATE class of requirements is necessary. 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 the evaluator uses to independently verify the vendor test results and to support of the test coverage analysis activities. ATE_COV.1 requires the developer to provide a test coverage analysis that demonstrates the extent to which the TSFI are tested by the developer's test suite. This component also requires an independent confirmation of the extent of the test suite, which aids in ensuring that correct security relevant functionality of a TSFI is demonstrated through the testing effort. 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 craft additional functional tests that address functional behavior that is not demonstrated in the developer's test suite. Upon successful completion of these requirements, the TOE's conformance to the specified security functional requirements will have been demonstrated.
O.RESIDUAL_ INFORMATION	FDP_RIP.1	FDP_RIP.1 is used to ensure the contents of

Objective	Requirements Addressing the Objective	Rationale
The TOE will ensure that any information contained in a protected resource within its Scope of Control is not released when the resource is reallocated.	FCS_CKM.4	resources are not available once the resource is reallocated. 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). FCS_CKM.4 requires the use of FIPS certification and places requirements on how cryptographic keys are managed within the TOE. This requirement places restrictions in addition to FDP_RIP.1, in that when a cryptographic key is moved from one location to another (e.g., calculated in some scratch memory and moved to a permanent location) that the memory area is immediately cleared as opposed to waiting until the memory is reallocated to another subject. FCS_CKM.4 also 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.
O.VULNERABILITY_ ANALYSIS The TOE will undergo some vulnerability analysis demonstrate the design and implementation of the TOE does not contain any obvious flaws.	AVA_VAN.2	The **AVA_VAN.2** component provides the necessary level of confidence that vulnerabilities do not exist in the TOE that could cause the security policies to be violated. AVA_VAN.2 requires the evaluator to perform a search for potential vulnerabilities in all the TOE deliverables. For those vulnerabilities that are not eliminated by the developer, a rationale must be provided that describes why these vulnerabilities cannot be exploited by a threat agent with a basic attack potential, which is in keeping with the desired assurance level of this TOE. This component provides the confidence that security flaws do not exist in the TOE that could be exploited by a threat agent of basic attack potential to violate the TOE's security policies. For this TOE, the vulnerability analysis is specified for an attack potential of basic. This requirement ensures the evaluator has performed an analysis of the authentication mechanism to ensure the probability of guessing a user's authentication data would require a medium-attack potential, as defined in Annex B of the CEM.

6.3 Rationale for the security objectives and security functional requirements for the environment

Table 15: Rationale for Requirements on the TOE IT Environment

Objective	Requirements Addressing the Objective	Rationale
OE.MANAGE The TOE IT environment will augment the TOE 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.	FAU_SAR.1 FAU_SAR.2 FAU_SAR.3 FIA_USB.1 FMT_MOF.1 FMT_MTD.1 FMT_SMR.1 FAU_SAA.1 FAU_STG.1 FAU_STG.3 FAU_GEN.2 FAU_SEL.1	FMT_SMR.1 ensures that the TOE IT environment provides an administrative role that may be used to manage both the TOE and the IT environment. FIA_USB.1 ensures that the TOE IT environment includes a mechanism to associate processes with roles. This ensures that both the TOE and its IT environment can identify its' associations. FAU_SAR.1 ensures that the IT environment provides those responsible for the TOE with facilities to review the TOE audit records (e.g., the Audit Administrator can construct a sequence of events provided the necessary events were audited). FAU_SAR.2 ensures that the TOE IT environment will be capable of limiting access to TOE audit records to only those with those users authorized to review them. FAU_SAR.3 provides the TOE's IT environment administrator with the ability to selectively review the contents of the audit trail based on established criteria. This capability allows the administrator to focus their audit review to what is pertinent at that time. FMT_MOF.1 ensures that the TOE IT environment limits access to TSF management functions to the administrator. FMT_MTD.1 ensures that the IT environment provides facilities to manage the time stamp mechanism. FAU_SAA.1 ensures that the IT environment monitors audited events base upon a set of rules that will indicate a potential violation to the administrator. FAU_STG.1 ensures that the IT environment will protect against the unauthorized deletion or modification of audit records. FAU_STG.3 provides that the administrator will be immediately alerted upon discovery of potential audit data loss. FAU_GEN.2 provides that the environment will be able to associate each event with the identity of the user that caused the event. This will allow the administrator to manage the audit data and monitor

Objective	Requirements Addressing the Objective	Rationale
		events associated with the user. FAU_SEL.1 allows the Security Administrator to configure which auditable events will be recorded the environment. 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.
OE.RESIDUAL_INFORMATION The TOE IT environment will ensure that any information contained in a protected resource within its Scope of Control is not released when the resource is reallocated.	FDP_RIP.1	FDP_RIP.1 ensures that the TOE IT environment provides same protections for residual information in a network packet that the TOE will provide. This ensures that neither the TOE nor the TOE IT environment will allow data from previously transmitted packets to be insert into new packets.
OE.SELF_PROTECTION The TOE IT environment will maintain a domain for itself and the TOE's own execution that protects them and their resources from external interference, tampering, or unauthorized disclosure through its their interfaces.	ADV_ARC.1	ADV_ARC.1 provides the security architecture description of the security domains maintained by the TSF that are consistent with the SFRs**.** Since self-protection is a property of the TSF that is achieved through the design of the TOE and TSF, and enforced by the correct implementation of that design, self-protection will be achieved by that design and implementation
OE.TIME_STAMPS The TOE IT environment shall provide reliable time stamps and the capability for the administrator to set the time used for these time stamps.	FPT_MTD.1 FPT_STM.1 FMT_SMR.1	FPT_STM.1 ensures that the IT environment provides a time stamp mechanism that may be used to synchronize audit events. FMT_MTD.1 ensures that the IT environment provides facilities to manage the time stamp mechanism and limits access to the time stamp mechanism to the administrator FMT_SMR.1 ensures that the TOE IT environment provides an administrative role that may be used to manage both the TOE and the IT environment.
OE.TOE_ACCESS The TOE IT environment will provide mechanisms that control a user's logical access to the TOE.	FMT_SMR.1 FIA_USB.1	FMT_SMR.1 ensures that the TOE IT environment provides an administrative role that may be used to manage both the TOE and the IT environment. FIA_USB.1 ensures that the TOE IT environment includes a mechanism to associate processes with roles. This ensures that both the TOE and its IT environment can identify its associations.

6.4 Additional Rationale for Security Objectives in the TOE IT Environment

Three of the security objectives for the TOE are simply restatements of an assumption found in Section 3. Therefore, these three objectives for the environment, OE.BASIC_ROBUSTNESS_OS, OE.NO_EVIL and OE.PHYSICAL, trace to the assumptions trivially.

Of these three, only OE.BASIC_ROBUSTNESS_OS bears further discussion. This assumption has been included in the PP because the TOE is not expected to address all of the threats and policies defined in a basic robustness environment. As such, the eventual user of the TOE must take additional steps to ensure the environment in which the TOE is used, has been hardened to the basic robustness level. This objective and its corresponding assumption should NOT be construed to allow the TOE IT environment to satisfy objectives or requirements levied on the TOE.

The remainder of the security objectives for the IT environment have been included in this Protection Profile in order to support the TOE security functions. The rationale support is documented in Table 13 along with the rationale for security objectives for the TOE.

6.5 Rationale for Assurance Requirements

EAL2 augmented was chosen to ensure a confidence in security services used to protect information in a Basic Robustness Environment. The assurance selection was based on:

- recommendations documented in the GIG guidance; and
- the postulated threat environment.

The EAL definitions in Part 3 of the CC were reviewed and the *Basic Robustness Assurance Package* (Evaluation Assurance Level (EAL) 2 augmented with, ALC_FLR.2 (Flaw Remediation.) was believed to best achieve this goal. The sponsor concluded that EAL2 augmented is applicable since this PP addresses circumstances where users require a basic level of independently assured security in commercial products. This level of assurance is commensurate with low threat environments or where compromise of protected information will not have a significant impact on mission objectives. This collection of assurance requirements requires TOE developers to gain assurance from good software engineering development practices, which do not require substantial specialist knowledge, skills, and other resources. Rationale for individual assurance requirements is provided in Table 14.

The Government's guidance for the GIG was consulted and found to also support the chosen assurance package.

The postulated threat environment specified in Section 3 of this PP was used in conjunction with the Information Assurance Technical Framework (IATF) Robustness Strategy guidance to derive the chosen assurance level.

These three factors were taken into consideration and the conclusion was that the basic robustness assurance package was the appropriate level of assurance.

6.6 Rationale for Not Satisfying All Dependencies

Each functional requirement, including extended requirements was analyzed to determine that all dependencies were satisfied. All requirements were then analyzed to determine that no additional dependencies were introduced as a result of completing each operation.

Table 16 identifies the functional requirement, its correspondent dependency and the analysis and rationale for not supporting the dependency in this PP.

Table 16: Unsupported Dependency Rationale

Requirement	Unsatisfied Dependencies	Dependency Analysis and Rationale
FDP_IFF.1	FMT_MSA.3	The FDP_IFF.1 requirement specifies the Wireless Client Policy. FMT_MSA.3 allows the PP author to specify secure default values for that policy. However, since the FMT_SMF.1(1) and FMT_SMF.1(3) provides the ability to set the policy, the ability to set a secure initial default value (e.g. decrypt by default) is not necessary.
FIA_USB.1	FIA_ATD.1	This dependency is on a requirement for the TOE IT environment. The purpose of requirements on the IT environment is to supplement the TOE and to ensure that the TOE and the IT environment together satisfy all security objectives. In order to limit the scope of the IT environment only those IT environmental requirements that directly contribute to the satisfaction of objectives have been included in this PP. Requirements for the IT environment that are necessary simply to satisfy management guidance, audit guidance or dependency chains have not been included in this PP. In the context of FIA_USB, the FIA_ATD dependency is used to specify user security attributes used to enforce the TSP. Since FIA_USB is specified for the TOE IT environment, FIA_ATD would also need to be specified for the TOE IT environment. However, including this requirement in the IT environment does not directly contribute to the satisfaction of any TOE objectives therefore it has been omitted.
FMT_SMR.1	FIA_UID.1	This dependency is on a requirement for the TOE IT environment. The purpose of requirements on the IT environment is to supplement the TOE and to ensure that the TOE and the IT environment together satisfy all security objectives. In order to limit the scope of the IT environment only those IT environmental requirements that directly contribute to the satisfaction of objectives have been included in this PP. Requirements for the IT environment that

Requirement	Unsatisfied Dependencies	Dependency Analysis and Rationale
		are necessary simply to satisfy management guidance, audit guidance or dependency chains have not been included in this PP. In the context of FMT_SMR the FIA_UID requirement is used to specify the action available to a user that has not been identified. It is expected that any role supported by the IT environment would require both identification and authentication components. However, including this requirement in the IT environment does not directly contribute to the satisfaction of any TOE objectives therefore it has been omitted.

6.7 Rationale for Extended requirements

Table 17 presents the rationale for the inclusion of the extended requirements found in this PP.

Table 17: Rationale for Extended Requirements

Extended Requirement	Identifier	Rationale
FAU_GEN_(EXT).1	Audit Data Generation	This extended requirement is necessary because the corresponding CC requirement (FAU_GEN.1) states that the TOE shall generate audit records indicating startup and shutdown of the audit log. This TOE is expected to generate audit records but it is not expected to control the audit log. Therefore it is not required to generate audit records for events related to startup and shutdown of the audit log.
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 implementation.
FCS_CKM_(EXT).2	Cryptographic Key Handling and Strorage	This extended requirement is necessary since the CC does not specifically provide components for key handling and storage.
FCS_COP_(EXT).1	Random number generation	This extended requirement is necessary since the CC cryptographic operation components are focused on specific algorithm types and operations requiring specific key sizes.

Extended Requirement	Identifier	Rationale
FPT_TST_(EXT).1	TSF Testing	This extended requirement is necessary to divide the TOE testing requirements between those necessary for the TOE itself and those specific to FIPS 140 cryptomodules.

7. References

1. US Government Wireless Access System for Basic Robustness Environments Protection Profile, Version *1.7.*

2. Common Criteria for Information Technology Security Evaluation, Version 3.1. CCMB-2006-09-001, 002, 003 . September 2006

3. Department of Defense Instruction 8500.2, "Information Assurance," February 6, 2003.

4. Common Methodology for Information Technology Security Evaluation, Version 3.1, CCMB-2006-09-004, September 2006.

5. FIPS PUB 140-2:Security Requirements for Cryptographic Modules, National Institute of Standards and Technology, May 25, 2001.

Appendix A. Acronyms

CC	Common Criteria
CM	Configuration Management
COTS	Commercial Off-The-Shelf
DoD	Department of Defense
EAL	Evaluation Assurance Level
FIPS	Federal Information Processing Standards
GIG	Global Information Grid
ISSE	Information System Security Engineers
IT	Information Technology
OSP	Organization Security Policy
PKI	Public Key Infrastructure
PP	Protection Profile
PUB	Publication
RF	Radio Frequency
SF	Security Function
SFP	Security Function Policy
SFR	Security Functional Requirement
ST	Security Target
TOE	Target of Evaluation
TSC	TSF Scope of Control
TSF	TOE Security Functions
TSFI	TSF Interface
TSP	TOE Security Policy
WLAN	Wireless Local Area Network

---

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**.