WatchGuard Technologies, Inc. Firebox® X Family: Core™ and Peak™ Series with Fireware™ Version 8.0 Security Target

Version: 2.1

Release Date: June 27, 2005

Prepared for:

WatchGuard Technologies, Inc.
505 5th Av. South Suite 500
Seattle WA, 98104
United States

Prepared by:

Corsec Security, Inc.
10340 Democracy Lane
Suite 201
Fairfax, VA 22030

Table of Contents

List of Tables

List of Figures

1 Security Target Introduction

This introductory section presents security target (ST) identification information and an overview of the ST structure. A brief discussion of the ST development methodology is also provided.

An ST document provides the basis for the evaluation of an information technology (IT) product or system (e.g., target of evaluation (TOE)). An ST principally defines:

- A set of assumptions about the security aspects of the environment;
- A list of threats which that product is intended to counter, and any known rules with which the product must comply; and
- A set of security objectives and a set of security requirements to address that problem.

The ST for a TOE is a basis for agreement between developers, evaluators, and consumers on a set of security properties of the TOE and the scope of the evaluation. The audience for an ST may include developers and "those responsible for managing, marketing, purchasing, installing, configuring, operating, and using the TOE", in addition to evaluators. As a result, this ST minimizes the use of terms from the *Common Criteria for Information Technology Security Evaluations* (CC).

An ST, like a Protection Profile (PP), contains sections which address the Security Environment, Security Objectives, and IT Security Requirements, as well as Security Objectives Rationale and Security Requirements Rationale sections. Under certain conditions, the contents of these sections of the ST may be identical with those of a PP, namely when the ST:

- Claims conformance with a PP;
- Performs no additional operations on the PP security functional requirements; and/or
- Does not extend the PP by adding security objectives and/or security requirements.

Under these conditions, the CC states that "reference to the PP is sufficient to define and justify the TOE objectives and requirements. Restatement of the PP contents is unnecessary".

The methodology used to develop and present this ST includes the following steps:

- Those PP security objectives and requirements with which the ST claims compliance and for which no additional operations are to be performed are restated within the ST.
- If the ST will perform additional operations on PP requirements, the ST restates the requirements, performs the operations, and identifies the change by convention.
- If the ST extends the PP by adding security objectives and/or security requirements, the ST states the objectives and/or requirements, makes any needed additions to the Security Environment section and documents suitable Rationale sections.

1.1 ST and TOE Identification

This section will provide information necessary to identify and control the Security Target and the TOE.

Table 1 - ST and TOE Description

ST Title:	WatchGuard Technologies, Inc. Firebox® X Family: Core™ and Peak™ Series with Fireware™ Version 8.0 Security Target Version 2.1 June 27, 2005
TOE Identification:	Firebox® X Family: Core™ and Peak™ Series with Fireware™ Version 8.0
CC Identification:	Common Criteria for Information Technology Security Evaluation, Version 2.1, Version 15408 FDIS, ISO/IEC SC27 N2161, Part 2 with international interpretations dated May 2004 CC Version 2.1 Part 2 with international interpretations dated May, 2004 – conformant CC Version 2.1 Part 3 with international interpretations dated May, 2004 – conformant
PP Identification:	U.S. Government Traffic -Filter Firewall Protection Profile for Low-Risk Environments Version 1.1 April 1999 (TFFPP)
Assurance Level:	Evaluation Assurance Level 4 (EAL 4)
Keywords:	Firewall, Traffic -Filter, Virtual Private Network, Router
ST Author	Corsec Security, Inc. 10340 Democracy Lane Suite 201 Fairfax, VA 22030

1.2 Security Target Overview

The Target of Evaluation is the Firebox® X Family: Core™ and Peak™ Series with Fireware™ Version 8.0. The hardware platform Core™ runs the firmware Fireware™ Version 8.0 (X500, X700, X1000, X2500 hardware configuration modules) and the Peak™ hardware platform runs the firmware Fireware™ Version 8.0 (X5000, X6000, and X8000 hardware configuration modules). The firmware Fireware™ Version 8.0 is build number 4057 and is the same firmware running on both the Core™ and Peak™ platforms. The WatchGuard Firebox® Core™ and Peak™ appliance will hereafter be referred to as "the Firebox® X Family", "Firebox® " or "the TOE".

This ST conforms to the U.S. Government Traffic -Filter Firewall Protection Profile for Low-Risk Environments. This particular Protection Profile requires assurance at Evaluation Assurance Level (EAL) 2. This ST has been augmented to meet the assurance requirements for EAL 4 while still meeting all of the functional requirements to conform to the specified Protection Profile.

The WatchGuard Firebox® ST contains the following sections:

Security Target Introduction: Presents the Security Target identification and an overview of the ST structure.

TOE Description: Provides an overview of the TOE security functions and describes the physical and logical boundaries for the TOE.

TOE Security Environment: Describes the threats and assumptions that pertain to the TOE and the TOE environment.

Security Objectives: Identifies the security objectives that are satisfied by the TOE and the TOE environment.

IT Security Requirements: Presents the Security Functional Requirements (SFRs) met by the TOE.

TOE Summary Specification: Describes the security functions provided by the TOE to satisfy the security requirements and objectives.

Protection Profile Claims: Presents the rationale concerning compliance of the ST to the TFFPP.

Rationale: Presents the rationale for the security objectives, requirements, and the TOE summary specifications as to their consistency, completeness, and suitability.

1.3 Common Criteria Conformance Claims

The TOE conforms to the U.S. Government Traffic-Filter Firewall Protection Profile for Low-Risk Environments, Version 1.1, FINAL.

It also conforms to Parts 2 and 3 of the CC, Version 15408 FDIS, ISO/IEC SC27 N2161.

This ST claims conformance to CC Version 2.1 Part 2 and 3 with international interpretations dated May 2004.

1.4 Conventions and Terminology

1.4.1 Conventions

The notation, formatting, and conventions used in this Security Target are largely consistent with those used in version 2 of the Common Criteria (CC) and the TFFPP. Selected presentation choices are discussed here to aid the Security Target user.

The CC allows several operations to be performed on security requirements; *refinement*, *selection assignment*, and *iteration* are defined in paragraph 2.1.4 of Part 2 of the CC. With the exception of the *iteration* operation, each of these operations is used in this Security Target.

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**. For example, see FMT_SMR.1 in this Security Target.

The **selection** operation is used to select one or more options provided by the CC in stating a requirement. Selections are denoted by *underlined italicized text*. For an example, see FDP_RIP.1 in this Security Target.

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]. For an example, see FDP_IFC.1 in this Security Target.

1.4.2 Terminology

In the Common Criteria, many terms are defined in Section 2.3 of Part 1. The following terms are a subset of those definitions. They are listed here to aid the user of the Security Target.

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

External IT entity – Any IT product or system, untrusted or trusted, outside of the TOE that interacts with the TOE.

Role – A predefined set of rules establishing the allowed interactions between a user and the TOE.

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

Authentication data – Information used to verify the claimed identity of a user.

In addition to the above general definitions, this Security Target provides the following specialized definitions:

Authorized Administrator – A role human users may be associated with in which to administer the security parameters of the TOE. Such users are not subject to any access control requirements once authenticated to the TOE and are therefore trusted to not compromise the security policy enforced by the TOE.

Authorized external IT entity – Any IT product or system, outside the scope of the TOE that may administer the security parameters of the TOE. Such entities are not subject to any access control requirements once authenticated to the TOE and are therefore trusted to not compromise the security policy enforced by the TOE.

1.4.3 Acronyms

The following abbreviations are used in this Security Target:

CC	Common Criteria for Information Technology Security Evaluation
CLI	Command Line Interface
DMZ	Demilitarized Zone
EAL	Evaluation Assurance Level
GUI	Graphical User Interface
HA	High Availability
HTTP	Hyper Text Transfer Protocol
IT	Information Technology
LED	Light Emitting Diode
PP	Protection Profile
SFP	Security Function Policy
SFR	Security Functional Requirement
SOF	Strength of Function
ST	Security Target
TFFPP	U.S. Government Traffic -Filter Firewall Protection Profile for Low-Risk Environments
TOE	Target of Evaluation
TSC	TSF Scope of Control
TSF	TOE Security Functions
TSP	TOE Security Policy

2 TOE Description

This section provides a general overview of the TOE in order to provide an understanding of how this TOE functions and to aid customers in determining whether this product meets their needs.

The Firebox® is designed to filter traffic coming through the Firebox® based on a set of rules that are created by a system administrator.

The Firebox® X Family is composed of two hardware platforms: Core™ series and Peak™ series. For each series, there is one physical appliance with upgradeable functionality that can be unlocked by purchasing a higher-end license feature key. The hardware configurations for the Core™ series are: X500, X700, X1000, and X2500. The hardware configurations for the Peak™ series are: X5000, X6000, and X8000.

The firmware version for all Firebox® X Family models is Fireware™ Version 8.0, build 4057.

2.1 Firebox® Core™ Hardware Platform

The WatchGuard Firebox® Core™ is a traffic filter firewall hardware appliance that is designed to run on a single platform, but with several hardware configurations.

The following hardware configurations are a part of this evaluation:

- X500
- X700
- X1000
- X2500

Each Firebox® is functionally the same as each of the others. The Core™ hardware platform is the same for all hardware configurations. Each hardware configurations add extra features, some of which are available outside the scope of this evaluation such as faster throughput and high availability. As the hardware configuration number (X500 through X2500) increases, there are more available interfaces and a higher bandwidth capacity.

All of the Firebox® appliances are run on a hardened Linux operating system that is based on Kernel version 2.4. All of the non-essential processes that are part of the Linux operating system have been removed, and there is no method of accessing the operating system directly. The only approved method of Firebox® administration is through the Command Line Interface (CLI).

2.2 Firebox® Peak™ Hardware Platform

The WatchGuard Firebox® Peak™ is a traffic filter firewall hardware appliance that is designed to run on a single platform, but with several hardware configurations.

The following hardware configurations are a part of this evaluation:

- X5000
- X6000
- X8000

Each Firebox® is functionally the same as each of the others. The Peak™ hardware platform is the same for all hardware configurations. Each hardware configurations add extra features, some of which are available outside the scope of this evaluation such as faster throughput and high availability. As the hardware configuration number (X5000 through X8000) increases, there are more available interfaces and a higher bandwidth capacity.

All of the Firebox® appliances are run on a hardened Linux operating system that is based on Kernel version 2.4. All of the non-essential processes that are part of the Linux operating system have been removed, and there is no method of accessing the operating system directly. The only approved method of Firebox® administration is through the CLI.

2.3 TOE Boundary

The TOE boundary is drawn as follows:

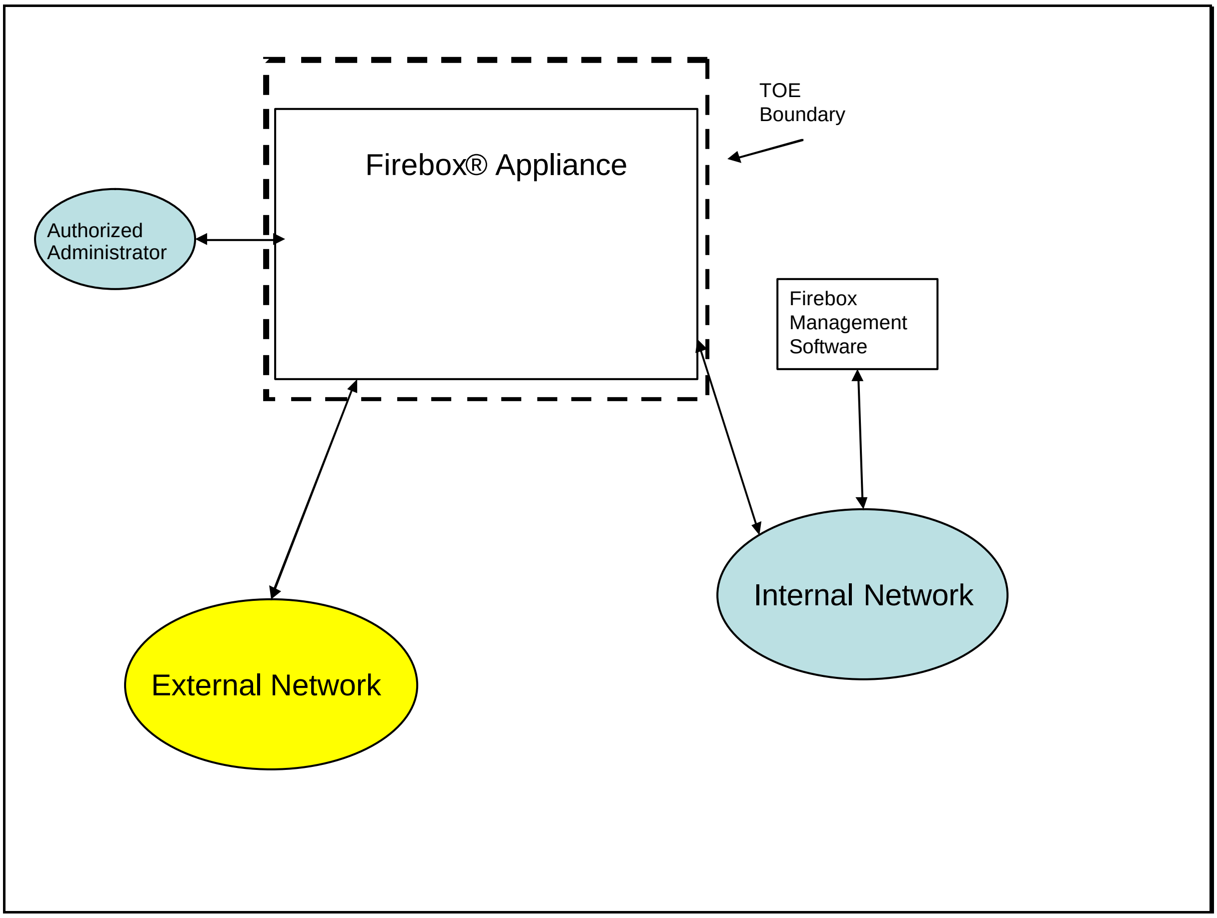

Figure 1 - TOE Boundary

The TOE Boundary is drawn around the WatchGuard Firebox® appliance. Access to administrative functions of the appliance is provided through a console port that utilizes a direct serial connection.

More detailed diagrams of the TOE Boundary are including in sections 2.4.3 Physical TOE Boundary and 2.5 Logical Scope.

2.4 Physical Scope

2.4.1 Firebox® Core™ Hardware Platform

The TOE boundary is drawn around the physical WatchGuard Firebox® Core™ appliance. Each Firebox® contains 6 RJ-45 Ethernet ports on the front of the chassis. Each Ethernet port has two Light Emitting Diodes (LEDs) to indicate if the connection is running at 10Mbps or 100Mbps. It is important to note that the functionality of these Ethernet ports is the same. All of the interfaces interact with the environment in the same way; however, each Firebox® model is designed for different network traffic demands.

All of the Firebox® appliances contain six (6) RJ-45 Ethernet ports that can be configured two different ways. These configurations are:

- Internal interface - used to connect to the internal network on which the Firebox® resides consisting of trusted subjects;
- External interface - used to connect to external networks that may be untrusted (i.e. the Internet)

The number of Ethernet ports that are active on a particular Firebox® is based on the model number. Although they may be physically different ports, they still interact with the TOE the exact same way.

Each Firebox® has a Console port that is used to access the Command Line Interface (CLI). The CLI is used by the administrator as a means of managing the Firebox® locally and is the only means of managing the Firebox®.

There are several parts of the Core™ that are either not security relevant or the parts are excluded from the evaluated configuration. They are:

- LCD Display
- Scrolling Buttons
- Removable Hard Drive Slot

The LCD Display is used for displaying the model number. The scrolling buttons are used to change the view of the LCD Display, but is not part of the evaluated configuration. The Removable hard drive slot is also not included as part of the evaluated configuration.

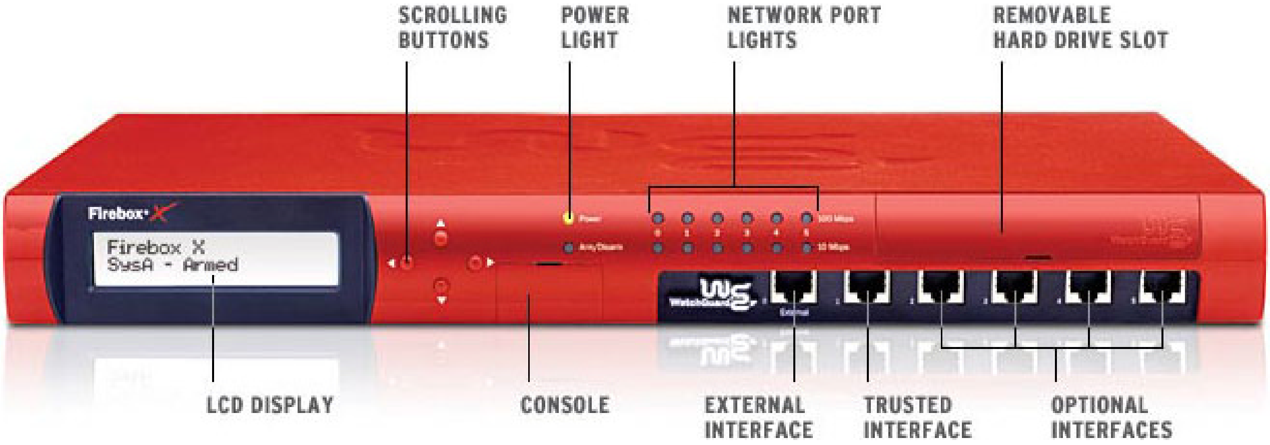

Figure 2 - View of Firebox® Core™ Hardware Platform

2.4.2 Firebox® Peak™ Hardware Platform

The Peak™ hardware platform has the same number of interfaces as the Core™ platform, except that the interfaces support Gigabit Ethernet. All of the other features are the same as described in the previous section.

2.4.3 Physical TOE Boundary

The following diagram shows how the physical components map into the TOE boundary. Although all of the Firebox® Core™ and Peak™ appliances have different hardware capabilities, they all have the same functionality as far as Firewall operation is concerned. Each of the Firebox® appliances has the options of having at least one internal and one external interface that is used to direct traffic from the external network to the internal network and vice versa. All of the traffic being received from and being transmitted to the external network goes through the external interface. All of the traffic being received from and being transmitted to the internal network is directed through the internal interface. The upper level Firebox® appliances have the capabilities to have an Ethernet port function as a Demilitarized Zone Interface (DMZ) or High Availability Interface (HA). This functionality is beyond the scope of this evaluation and have been excluded as part of the TOE.

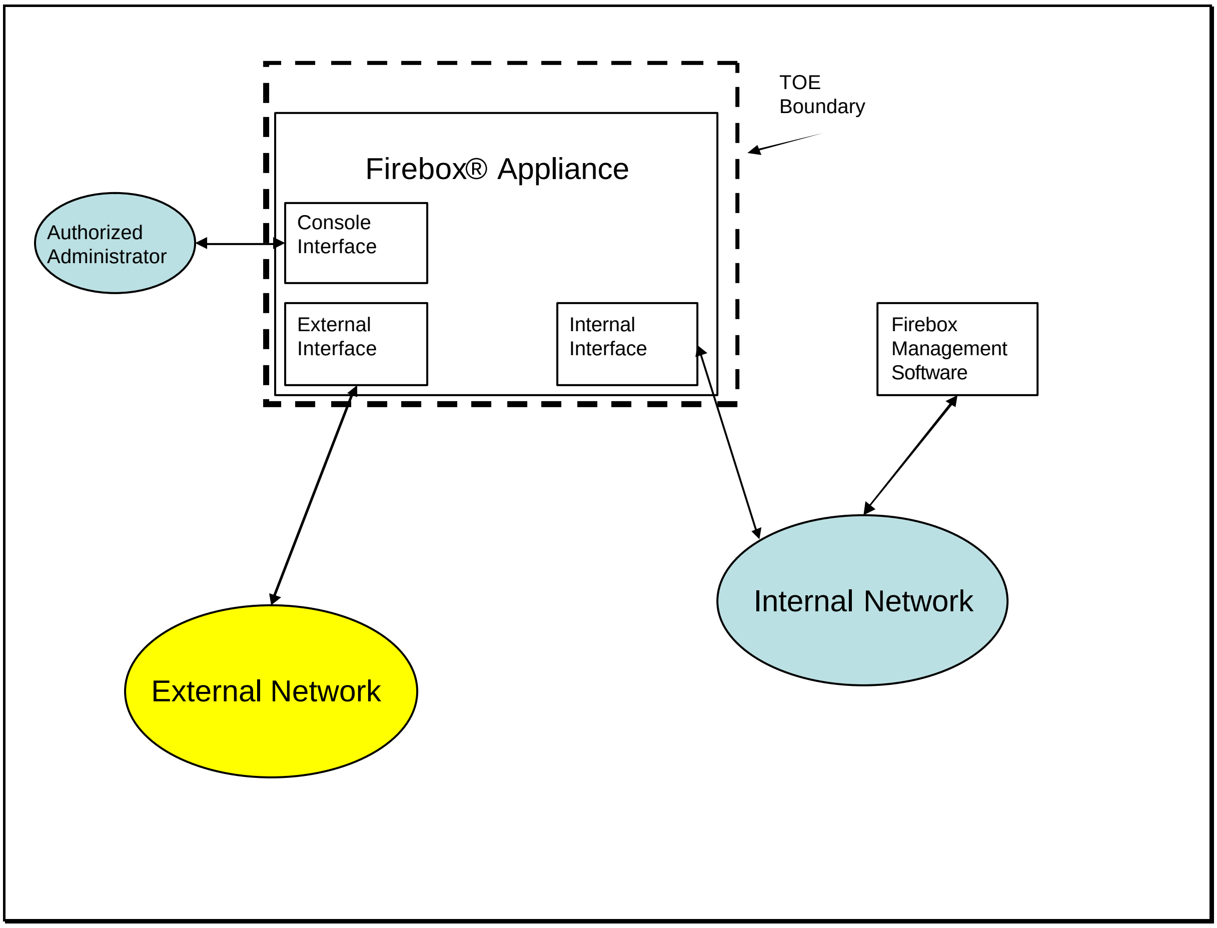

Figure 3 - TOE Physical Boundary

2.5 Logical Scope

The Logical Boundary of the TOE encompasses all of the software components that are run from the physical Firebox® appliance. All of the Firebo x® components run on top of a Linux Kernel v. 2.4.

All of the Firewall policy enforcement is performed by the RapidCore engine (RCE) software that WatchGuard designed specifically for Firebox® appliances. The following diagram shows how the RapidCore eng ine interfaces with the other logical Firebox® components.

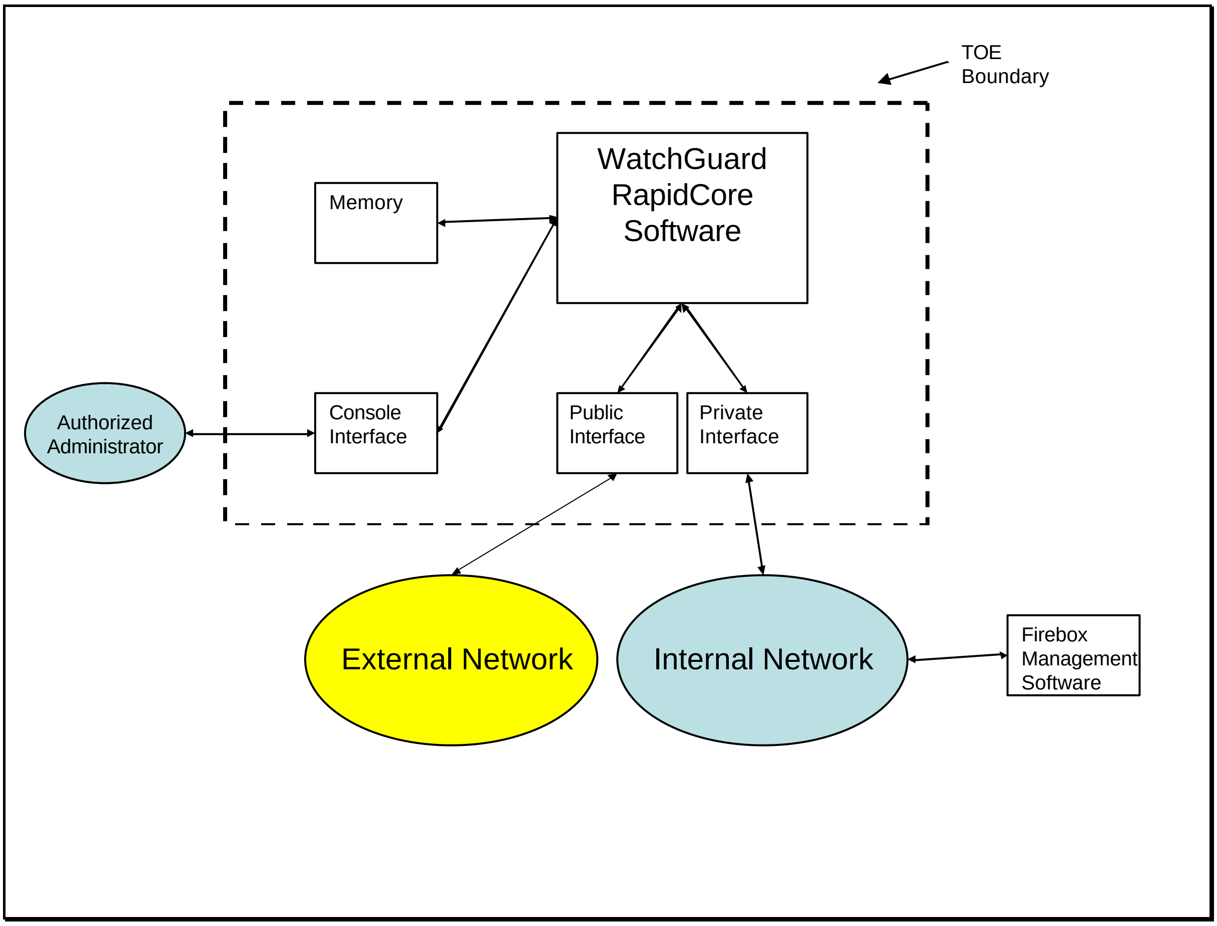

Figure 4 - Logical TOE Boundary

The RapidCore engine handles all of the information flow and interfaces with the memory on the Firebox® to store user account information, audit records, and policy information.

2.6 Evaluated Configuration

The evaluated configuration of the WatchGuard Firebox® is intended to use the appliance as a Firewall. The evaluated configuration will be limited to administration through the CLI only and will not include VPN, DMZ, or High Availability functionality.

3 TOE Security Environment

This section aims to clarify the nature of the security problem that the WatchGuard Firebox® X Family appliances are intended to solve. It does so by describing:

- Any assumptions about the security aspects of the environment and/or of the manner in which the WatchGuard Firebox® X Family appliances are intended to be used.

- Any known or assumed threats to the assets against which specific protection within the WatchGuard Firebox® X Family appliances or their environment is required.

The WatchGuard Firebox® X Family appliances are intended to be used either in environments in which, at most, sensitive but unclassified information is processed, or the sensitivity level of information in both the internal and external networks is equivalent.

3.1 Assumptions

This section helps define the scope of the security problem by identifying assumptions about the security aspects of the environment and/or of the manner in which the WatchGuard Firebox® X Family appliances are intended to be used.

3.1.1 Assumptions from the TFFPP

The TOE claims all the assumptions delineated below within the TFFPP. Those assumptions that are claimed are stated below (with occasional minor modifications to correct grammatical or other incidental errors present in the TFFPP).

A.PHYSEC The TOE is physically secure.

A.LOWEXP The threat of malicious attacks aimed at discovering exploitable vulnerabilities is considered low.

A.GENPUR There are no general-purpose computing capabilities (e.g., the ability to execute arbitrary code or applications) and storage repository capabilities on the TOE other than to support the TSF.

A.PUBLIC The TOE does not host public data.

A.NOEVIL Authorized administrators are non-hostile and follow all administrator guidance; however, they are capable of error.

A.SINGEN Information cannot flow among the internal and external networks unless it passes through the TOE.

A DIRECT Human users within the physically secure boundary protecting the TOE may attempt to access the TOE from some direct connection (e.g., a console port) if the connection is part of the TOE.

3.2 Threats

This section helps define the nature and scope of the security problem by identifying assets which require protection as well as threats to those assets.

Threats may be addressed either by the WatchGuard Firebox® X Family appliances or by their intended environment (for example, using personnel, physical, or administrative safeguards). These two classes of threats are discussed separately.

3.2.1 Threats to be Addressed by the TOE

The TOE addresses all threats delineated below from the TFFPP. These threats are restated from the TFFPP.

T.NOAUTH An unauthorized person may attempt to bypass the security of the TOE so as to assess and use security functions and/or non-security functions provided by the TOE.

T.REPEAT An unauthorized person may repeatedly try to guess authentication data in order to use this information to launch attacks on the TOE.

T.REPLAY An unauthorized person may use valid identification and authentication data obtained to access functions provided by the TOE.

T.ASPOOF An unauthorized person may carry out spoofing in which information flow through the TOE into a connected network by using a spoofed source address.

T.MEDIAT An unauthorized person may send impermissible information through the TOE that result in the exploitation of resources on the internal network.

T.OLDINF Because of a flaw in the TOE functioning, an unauthorized person may gather residual information from a previous information flow or internal TOE data by monitoring the padding of the information flows from the TOE.

T.AUDACC Persons may not be accountable for the actions that they conduct because the audit records are not reviewed, thus allowing an attacker to escape detection.

T.SELPRO An unauthorized person may read, modify, or destroy security critical TOE configuration data.

T.AUDFUL An unauthorized person may cause audit records to be lost or prevent future records from being recorded by taking actions to exhaust audit storage capacity, thus masking an attacker's actions.

3.2.2 Threats to Be Addressed by the Environment

The threat possibility discussed below must be countered by procedural measures and/or administrative methods.

T.TUSAGE The TOE may be inadvertently configured, used and administered in an insecure manner by either authorized or unauthorized persons.

4 Security Objectives

The purpose of the security objectives is to detail the planned response to a security problem or threat. Threats can be directed against the TOE or the TOE operating environment or both; therefore, the CC identifies two categories of security objectives:

- Security objectives of the TOE, and
- Security objectives for the Operating Environment.

4.1 Security Objectives for the TOE

The following are the IT security objectives for the TOE:

O.IDAUTH The TOE must uniquely identify and authenticate the claimed identity of all users, before granting a user access to TOE functions.

O.SINUSE The TOE must prevent the reuse of authentication data for users attempting to authenticate at the TOE from a connected network.

O.MEDIAT The TOE must mediate the flow of all information from users on a connected network to users on another connected network, and must ensure that residual information from a previous information flow is not transmitted in any way.

O.SECSTA Upon initial start-up of the TOE or recovery from an interruption in TOE service, the TOE must not compromise its resources or those of any connected network.

O.SELPRO The TOE must protect itself against attempts by unauthorized users to bypass, deactivate, or tamper with TOE security functions.

O.AUDREC The TOE must provide a means to record a readable audit trail of security-related events, with accurate dates and times, and a means to search and sort the audit trail based on relevant attributes.

O.ACCOUN The TOE must provide user accountability for information flows through the TOE and for authorized administrator use of security functions related to audit.

O.SECFUN The TOE must provide functionality that enables an authorized administrator to use the TOE security functions, and must ensure that only authorized administrators are able to access such functionality.

O.LIMEXT The TOE must provide the means for an authorized administrator to control and limit access to TOE security functions by an authorized external IT entity.

4.2 Security Objectives for the TOE Environment

The following security objectives for the TOE environment are derived from the assumptions stated in the TFFPP.[1]

All of the assumptions stated in Section 3.1.1 are considered to be security objectives for the environment. The following are the Security Target non-IT security objectives, which, in addition to those assumptions, are to be satisfied without imposing technical requirements on the TOE. That is, they will not require the implementation of functions in the TOE hardware and/or software. Thus, they will be satisfied largely through application or procedural or administrative measures.

O. PHYSEC The TOE is physically secure.

O.LOWEXP The threat of malicious attacks aimed at discovering exploitable vulnerabilities is considered low.

O.GENPUR There are no general-purpose computing capabilities (e.g. the ability to execute arbitrary code or applications) and storage repository capabilities on the TOE other than to support the TSF.

O.PUBLIC The TOE does not host public data.

O.NOEVIL Authorized administrators are non-hostile and follow all administrator guidance; however, they are capable of error.

O.SINGEN Information can not flow among the internal and external networks unless it passes through the TOE.

O.DIRECT Human users within the physically secure boundary protecting the TOE may attempt to access the TOE from some direct connection (e.g. a console port) if the connection is part of the TOE.

O.GUIDAN The TOE must be delivered, installed, administered, and operated in a manner that maintains security.

O.ADMTRA Authorized administrators are trained as to establishment and maintenance of security policies and practices.

4.3 Organizational Security Policies

This section addresses any OSP that the WatchGuard Firebox® must comply to.

The TFFPP states one OSP relating to the use of cryptographic modules. Because this TOE is not providing remote administration, this OSP does not apply. Therefore, no organizational security policy is specified.

[1] The first nine (9) security objectives for the IT environment were changed from the PP. The PP had listed the Objectives proceeded by an A. The objectives have been changed to be proceeded by an O to match the security objectives for the TOE in the previous section.

5 IT Security Requirements

IT security requirements include:

- TOE security requirements; and (optionally)
- Security requirements for the TOE's IT environment (that is, for hardware, software, or firmware external to the TOE and upon which satisfaction of the TOE's security objectives depends).

These requirements are discussed separately below.

5.1 TOE Security Requirements

The CC divides security requirements into two categories:

- Security functional requirements (SFRs), that is, requirements for security functions such as information flow control, audit, identification and authentication.
- Security assurance requirements (SARs) which provide grounds for confidence that the TOE meets its security objectives (for example, configuration management, testing, vulnerability assessment).

This section presents the SFRs for the TOE. The SARs are listed in section 6.

5.1.1 TOE Security Functional Requirements

This section presents the restated SFRs for the TOE.

The TOE shall satisfy the SFRs stated in the table below which lists the CC names of the SFR components contained in the TFFPP. Following the table, the individual functional requirements are restated from the TFFPP.

Table 2 - Restated Security Functional Requirements

Functional Requirement ID	Functional Requirement Name
FMT_SMR.1	Security roles
FIA_AFL.1	Authentication failure handling
FIA_ATD.1	User attribute definition
FIA_UID.2	User identification before any action
FIA_UAU.2	User authentication before any action
FDP_IFC.1	Subset information flow control
FDP_IFF.1	Simple security attributes
FMT_MSA.3	Static attribute initialization
FDP_RIP.1	Subset residual information protection
FPT_RVM.1	Non-bypassability of the TSP
FPT_SEP.1	TSF domain separation

Functional Requirement ID	Functional Requirement Name
FPT_STM.1	Reliable time stamps
FAU_GEN.1	Audit data generation
FAU_SAR.1	Audit review
FAU_SAR.3	Selectable audit review
FAU_STG.1	Protected audit trail storage
FAU_STG.4	Prevention of audit data loss
FMT_MOF.1	Management of security functions behavior

Additional requirements have been included that were not part of the TFFPP. These requirements are listed in the following table.

Table 3 – Augmented Security Functional Requirements

Functional Requirement ID	Functional Requirement Name
FMT_SMF.1	Specification of Management Functions

The statement of the TOE security requirements must include a minimum strength level for the TOE security functions realized by a probabilistic or permutational mechanism. In the case of this Security Target, the minimum level of strength shall be SOF-Basic. For a rationale for this selected level, see Section 9.4.

Specific strength of function metrics are defined for the following requirements:

> **FIA_UAU.2** – Strength of Function shall be demonstrated such that the probability that authentication data can be guessed is no greater than one in one million.

The following paragraphs are intended to clarify why the functional components in this Security Target are presented in the order outlined in Table 2. FMT_SMR.1 is the first component listed because it defines the authorized administrator role, which appears in a number of the components that follow.

The class FIA components are listed after FMT_SMR.1. They describe the identification and authentication policy that all users, both human users and external IT entities, must abide by before being able to use other TOE functions.

The order of the class FIA components was chosen on the following basis. Since users are already defined in the Terminology section on page 8, the Security Target reader is introduced in component FIA_ATD.1 to those users' security attributes. The next component, FIA_UID.2, forces users to identify themselves to the TOE user the user security attributes of component FIA_ATD.1 before further actions take place. Since authentication must follow successful identification, component FIA_UAU.2 appears after FIA_UID.2. Then, component FIA_AFL.1 describes what results if the user fails to authenticate after some settable number of attempts.

There is an information flow control SFP, and it is defined after the class FIA components in FDP_IFC.1. The policy rules which must be enforced as well as the attributes of the entities defined in FDP_IFC.1 are then written in FDP_IFF.1. Component FMT_MSA.3, which FDP_IFF.1 depends on, follows. As part of the installation and start-up of the TOE, FMT_MSA.3 mandates a default deny policy which permits no information to flow through the TOE. FDP_RIP.1 is listed next, ensuring that resources are cleared before being allocated to hold packets of information at the TOE.

Components dealing with the protection of trusted security functions come next. These include components FPT_RVM.1 and FPT_SEP.1.

Since FAU_GEN.1 requires recording the time and date when audit events occur, it follows the FPT_STM.1 component that alerts developers that an accurate time and date must be maintained on the TOE. The class FAU requirements follow to define the audit security functions which must be supported by the TOE. FAU_GEN.1 is the first audit component listed because it depicts all the events that must be audited, including all the information which must be recorded in audit records. The remainder of the class FAU components ensure that the audit records can be read (component FAU_SAR.1), searched and sorted (component FAU_SAR.3, and protected from modification (FAU_STG.1). Lastly, FAU_STG.4 ensures that the TOE is capable of preventing auditable actions, not taken by an authorized administrator, from occurring in the event that the audit trail becomes full.

The last component in the Security Target is FMT_MOF.1. It appears last because it lists all the functions to be provided by the TOE for use only by the authorized administrator. Almost all of these functions are based on components which precede it. Thus it is listed last.

FMT_SMR.1 Security roles

FMT_SMR.1.1 The TSF shall maintain the roles [authorized administrator].

FMT_SMR.1.2 The TSF shall be able to associate **human** users with **the authorized administrator** roles.

FIA_ATD.1 User attribute definition

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

a) [Identity;

b) Association of a human user with the authorized administrator role;

c) A password to confirm the identity of the user].

FIA_UID.2 User identification before any action

FIA_UID.2.1 The TSF shall require each user to identify itself before allowing any other TSF-mediated actions on behalf of that user.

FIA_UAU.2 User authentication before any action

FIA_UAU.2.1 The TSF shall require each **Authorized Administrator** to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

FIA_AFL.1 Authentication failure handling

FIA_AFL.1.1 The TSF shall detect when [3] unsuccessful authentication attempts occur related to [external IT entities attempting to authenticate from an internal or external network].

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF shall [prevent the offending external IT entity from successfully authenticating until an authorized administrator takes some action to make authentication possible for the external IT entity in question].

FDP_IFC.1 Subset information flow control

FDP_IFC.1.1 The TSF shall enforce the [UNAUTHENTICATED SFP] on:

a) [Subjects: unauthenticated external IT entities that send and receive information through the TOE to one another;

b) Information: traffic sent through the TOE from one subject to another;

c) Operation: pass information].

FDP_IFF.1 Simple security attributes[2]

FDP_IFF.1.1 The TSF shall enforce the [UNAUTHENTICATED SFP] based on **at least** the following types of subject and information security attributes:

- [Subject security attributes:
  - presumed address;

- Information security attributes
  - Presumed address of source subject;
  - Presumed address of destination subject;
  - Transport layer protocol;
  - TOE interface on which traffic arrives and departs;
  - Service;

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

a) [Subjects on an internal network can cause information to flow through the TOE to another connected network if:
   - All the information security attribute values are unambiguously permitted by the information flow security policy rules, where such rules may be composed from

[2] The complete set of functional elements of a component must be selected for inclusion in a ST. However, since the following functional elements from the FDP_IFF.1 component do not add anything significant to the ST, they have been moved here to allow for a clearer, smoother flowing presentation of the FDP_IFF.1
FDP_IFF.1.3 – The TSF shall enforce the [none].
FDP_IFF.1.4 – The TSF shall provide the following [none].
FDP_IFF.1.5 – The TSF shall explicitly authorize an information flow based on the following rules: [none].

all possible combinations of the values of the information flow security attributes, created by the authorized administrator;

- The presumed address of the source subject, in the information, translates to an internal network address;
- And the presumed address of the destination subject, in the information, translates to an address on the other connected network.

b) Subjects on the external network can cause information to flow through the TOE to another connected network if:

- All the information security attribute values are unambiguously permitted by the information flow security policy rules, where such rules may be composed from all possible combinations of the values of the information flow security attributes, created by the authorized administrator;
- The presumed address of the source subject, in the information, translates to an external network address;
- And the presumed address of the destination subject, in the information, translates to an address on the other connected network.]

FDP_IFF.1.6 The TSF shall explicitly deny an information flow based on the following rules:

a) [The TOE shall reject requests for access or services where the information arrives on an external TOE interface, and the presumed address of the source subject is an external IT entity on an internal network;

b) The TOE shall reject requests for access or services where the information arrives on an internal TOE interface, and the presumed address of the source subject is an external IT entity on the external network;

c) The TOE shall reject requests for access or services where the information arrives on either an internal or external TOE interface, and the presumed address of the source subject is an external IT entity on a broadcast network;

d) The TOE shall reject requests for access or services where the information arrives on either an internal or external TOE interface, and the presumed address of the source subject is an external IT entity on the loopback network.]

FMT_MSA.3 Static attribute initialization

FMT_MSA.3.1 The TSF shall enforce the [UNAUTHENTICATED SFP] to provide *restrictive* default values for **information flow** security attributes that are used to enforce the SFP.

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

FDP_RIP.1 Subset residual information protection

FDP_RIP.1.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the *allocation of the resource to* the following objects: [resources that are used by the subjects of the TOE to communicate through the TOE to other subjects].

FPT_RVM.1 Non-bypassability of the TSP

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

FPT_SEP.1 TSF domain separation

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

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

FPT_STM.1 Reliable time stamps

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

FAU_GEN.1 Audit data generation

FAU_GEN.1.1 The TSF shall be able to generate an audit record of the following auditable events:

a) Start-up and shutdown of the audit functions;

b) All **relevant** auditable events for the *minimal or basic* level of audit **specified in** Table 4**;** and

c) [the event in Table 4 listed at the "extended" level].

FAU_GEN.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**s** identit**ies**, 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 ST, [information specified in column four of Table 4].

Table 4 – Auditable Events

Functional Component	Level	Auditable Event	Additional Audit Record Contents
FMT_SMR.1	Minimal	Modifications to the group of users that are part of [the authorized administrator] role.	The identity of the authorized administrator performing the modification and the user identity being associated with the authorized administrator role.
FIA_UID.2	Basic	All use of the user identification mechanism	The user identities provided to the TOE
FIA_UAU. 2	Basic	All use of the authentication mechanism.	The user identities provided to the TOE.
FIA_AFL.1	Minimal	The reaching of the threshold for unsuccessful authentication attempts and the subsequent [restoration by the authorized administrator of the users' capability to authenticate.]	The identity of the offending user and the authorized administrator.
FDP_IFF.1	Basic	All decisions on requests for information flow.	The presumed addresses of the source and destination subject.
FPT_STM.1	Minimal	Changes to the time.	The identity of the authorized administrator performing the operation.
FMT_SMF.1	Minimal	Whenever policy rules are created, modified, edited, or deleted	The authorized administrator's role.
FMT_MOF.1	Extended	Use of the functions listed in this requirement pertaining to audit.	The identity of the authorized administrator performing the operation.

FAU_SAR.1 Audit Review

FAU_SAR.1.1 The TSF shall provide [an authorized administrator] with the capability to read [all audit trail data] from the audit records.

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

FAU_SAR.3 Selectable audit review

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

a) [Presumed subject address;

b) ranges of dates;

c) ranges of times; and

d) ranges of addresses].

FAU_STG.1 Protected audit trail storage

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

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

FAU_STG.4 Prevention of audit data loss

FAU_STG.4.1 The TSF shall *prevent auditable events, except those taken by the authorized* ***administrator*** and [shall limit the number of audit records lost] if the audit trail is full.

FMT_SMF.1 Specification of Management Functions

FMT_SMF.1.1 The TSF shall be capable of performing the following security management functions:[Create, view, modify, or delete policy rules to either permit or block network traffic being transmitted through the TOE].

FMT_MOF.1 Management of security functions behavior

FMT_MOF.1.1 The TSF shall restrict the ability to *modify the behavior of* the functions:

a) [Start-up and shutdown;

b) Create, delete, modify, and view information flow security policy rules that permit or deny information flows;

c) Create, delete, modify, and view user attribute values defined in FIA_ATD.1;

d) Enable and disable external IT entities from communicating to the TOE (if the TOE supports authorized external IT entities);

e) Modify and set the time and date;

f) Archive, create, delete, empty, and review the audit trail;

g) Backup of user attribute values, information flow security policy rules, and audit trail data, where the backup capability shall be supported by automated tool;

h) Recover to the state following the last backup.]

to [an authorized administrator].

6 TOE Security Assurance Requirements

The table below identifies the security assurance components drawn from CC Part 3: Security Assurance Requirements, EAL 4 that apply to the TOE. Assurance requirements have been taken from Section 5.1.2, TOE Security Assurance Requirements of the TFFPP. The additional assurance requirements required for EAL 4 that were not included in the TFFPP have been taken from Part 3 of the CC. The assurance level for this evaluation is level 4 which is an augmentation from the assurance requirements contained in the PP. The assurance requirements that were augmented from the PP are marked with an asterisk (*) in the following table.

Table 5 – EAL 4 SARs

Assurance Component ID	Assurance Component Name
ACM_AUT.1*	Partial CM Automation
ACM_CAP.4*	Generation support and acceptance procedures
ACM_SCP.2*	Problem tracking CM coverage
ADO_DEL.2*	Detection of modification
ADO_IGS.1	Installation, generation, and start-up procedures
ADV_FSP.2*	Fully defined external interfaces
ADV_HLD.2*	Security enforcing high-level design
ADV_IMP.1*	Subset of the implementation of the TSF
ADV_LLD.1*	Descriptive low-level design
ADV_RCR.1	Informal correspondence demonstration
ADV_SPM.1*	Informal TOE security policy model
AGD_ADM.1	Administrator guidance
AGD_USR.1	User guidance
ALC_DVS.1*	Identification of security measures
ALC_LCD.1*	Developer defined life-cycle model
ALC_TAT.1*	Well-defined development tools
ATE_COV.2*	Analysis of coverage
ATE_DPT.1*	Testing: high-level design
ATE_FUN.1	Functional testing
ATE_IND.2	Independent testing – sample
AVA_MSU.2*	Validation of Analysis
AVA_SOF.1	Strength of TOE security function evaluation
AVA_VLA.2*	Independent vulnerability analysis

6.1.1 ACM_AUT.1 Partial CM Automation

Developer action elements:

ACM_AUT.1.1D The developer shall use a CM system.

ACM_AUT.1.2D The developer shall provide a CM plan.

Content and presentation of evidence elements:

ACM_AUT.1.1C The CM system shall provide an automated means by which only authorized changes are made to the TOE implementation representation.

ACM_AUT.1.2C The CM system shall provide an automated means to support the generation of the TOE.

ACM_AUT.1.3C The CM plan shall describe the automated tools used in the CM system.

ACM_AUT.1.4C The CM plan shall describe how the automated tools are used in the CM system.

6.1.2 ACM_CAP.4 Generation support and acceptance procedures

Developer action elements:

ACM_CAP.4.1D The developer shall provide a reference for the TOE.

ACM_CAP.4.2D The developer shall use a CM system.

ACM_CAP.4.3D The developer shall provide CM documentation.

Content and presentation of evidence elements:

ACM_CAP.4.1C The reference for the TOE shall be unique to each version of the TOE.

ACM_CAP.4.2C The TOE shall be labeled with its reference.

ACM_CAP.4.3C The CM documentation shall include a configuration list, a CM plan, and an acceptance plan.

ACM_CAP.4.4C The configuration list shall describe the configuration items that comprise the TOE.

ACM_CAP.4.5C The CM documentation shall describe the method used to uniquely identify the configuration items.

ACM_CAP.4.6C The CM system shall uniquely identify all configuration items.

ACM_CAP.4.7C The CM plan shall describe how the CM system is used.

ACM_CAP.4.8C The evidence shall demonstrate that the CM system is operating in accordance with the CM plan.

ACM_CAP.4.9C The CM documentation shall provide evidence that all configuration items have been and are being effectively maintained under the CM system.

ACM_CAP.4.10C The CM system shall provide measures such that only authorized changes are made to the configuration items.

ACM_CAP.4.11C	The CM system shall support the generation of the TOE.
ACM_CAP.4.12C	The acceptance plan shall describe the procedures used to accept modified or newly created configuration items as part of the TOE.

6.1.3 ACM_SCP.2 Problem tracking CM coverage

Developer action elements:

ACM_SCP.2.1D	The developer shall provide CM documentation.

Content and presentation of evidence elements:

ACM_SCP.2.1C	The CM documentation shall show that the CM system, as a minimum, tracks the following: the TOE implementation representation design documentation, test documentation, user documentation, administrator documentation, and CM documentation and security flaws.
ACM_SCP.2.2C	The CM documentation shall describe how configuration items are tracked by the CM system.

6.1.4 ADO_DEL.2 Detection of modification

Developer action elements:

ADO_DEL.2.1D	The developer shall document procedures for delivery of the TOE or parts of it to the user.
ADO_DEL.2.2D	The developer shall use the delivery procedures.

Content and presentation of evidence elements:

ADO_DEL.2.1C	The delivery documentation shall describe all procedures that are necessary to maintain security when distributing versions of the TOE to a user's site.
ADO_DEL.2.2C	The delivery documentation shall describe how the various procedures and technical measures provide for the detection of modifications, or any discrepancy between the developer's master copy and the version received at the user site.
ADO_DEL.2.3C	The delivery documentation shall describe how the various procedures allow detection of attempts to masquerade as the developer, even in cases in which the developer has sent nothing to the user's site.

6.1.5 ADO_IGS.1 Installation, generation, and start-up procedures

Developer action elements:

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

Content and presentation of evidence elements:

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

6.1.6 ADV_FSP.2 Fully defined external interfaces

Developer action elements:

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

Content and presentation of evidence elements:

ADV_FSP.2.1C The functional specification shall describe the TSF and its external interfaces using an informal style.

ADV_FSP.2.2C The functional specification shall be internally consistent.

ADV_FSP.2.3C The functional specification shall describe the purpose and method of use of all external TSF interfaces, providing complete details of all effects, exceptions and error messages.

ADV_FSP.2.4C The functional specification shall completely represent the TSF.

ADV_FSP.2.5C The functional specification shall include rationale that the TSF is completely represented.

6.1.7 ADV_HLD.2 Security enforcing high-level design

Developer action elements:

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

Content and presentation of evidence elements:

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

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

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

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

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

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

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

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

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

6.1.8 ADV_IMP.1 Subset of the implementation of the TSF

Developer action elements:

ADV_IMP.1.1D The developer shall provide the implementation representation for a selected subset of the TSF.

Content and presentation of evidence elements:

ADV_IMP.1.1C The implementation representation shall unambiguously define the TSF to a level of detail such that the TSF can be generated without further design decisions.

ADV_IMP.1.2C The implementation representation shall be internally consistent

6.1.9 ADV_LLD.1 Descriptive low level design

Developer action elements:

ADV_LLD.1.1D The developer shall provide the low-level design of the TSF.

Content and presentation of evidence elements:

ADV_LLD.1.1C The presentation of the low-level design shall be informal.

ADV_LLD.1.2C The low-level design shall be internally consistent.

ADV_LLD.1.3C The low-level design shall describe the TSF in terms of modules.

ADV_LLD.1.4C The low-level design shall describe the purpose of each module.

ADV_LLD.1.5C The low-level design shall define the interrelationships between the modules in terms of provided security functionality and dependencies on other modules.

ADV_LLD.1.6C The low-level design shall describe how each TSP -enforcing function is provided.

ADV_LLD.1.7C The low-level design shall identify all interfaces to the modules of the TSF.

ADV_LLD.1.8C The low-level design shall identify which of the interfaces to the modules of the TSF are externally visible.

ADV_LLD.1.9C The low-level design shall describe the purpose and method of us of all interfaces to the modules of the TSF, providing details of effects, exceptions and error messages, as appropriate.

ADV_LLD.1. 10C The low-level design shall describe the separation of the TOE into TSP-enforcing and other modules.

6.1.10 ADV_RCR.1 Informal correspondence demonstration

Developer action elements:

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

Content and presentation of evidence elements:

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

6.1.11 ADV_SPM.1 Informal TOE security policy model

Developer action elements:

ADV_SPM.1.1D The developer shall provide a TSP model.

ADV_SPM.1.2D The developer shall demonstrate correspondence between the functional specification and the TSP model.

Content and presentation of evidence elements:

ADV_SPM.1.1C The TSP model shall be informal.

ADV_SPM.1.2C The TSP model shall describe the rules and characteristics of all policies of the TSP that can be modeled.

ADV_SPM.1.3C The TSP model shall include a rationale that demonstrates that it is consistent and complete with respect to all policies of the TSP that can be modeled.

ADV_SPM.1.4C The demonstration of correspondence between the TSP model and the functional specification shall show that all of the security functions in the functional specification are consistent and complete with respect to the TSP model.

6.1.12 AGD_ADM.1 Administrator guidance

Developer action elements:

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

Content and presentation of evidence elements:

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

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

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

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

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

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

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

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

6.1.13 AGD_USR.1 User guidance

Developer action elements:

AGD_USR.1.1D The developer shall provide user guidance.

Content and presentation of evidence elements:

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

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

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

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

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

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

6.1.14 ALC_DVS.1 Identification of security measures

Developer action elements:

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

Content and presentation of evidence elements:

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

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

6.1.15 ALC_LCD.1 Developer defined life-cycle model

Developer action elements:

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

ALC_LCD.1.2D The developer shall provide life-cycle definition documentation.

Content and presentation of evidence elements:

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

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

6.1.16 ALC_TAT.1 Well-defined development tools

Developer action elements:

ALC_TAT.1.1D The developer shall identify the development tools being used for the TOE.

ALC_TAT.1.2D The developer shall document the selected implementation-dependent options of the development tools.

Content and presentation of evidence elements:

ALC_TAT.1.1C All development tools used for implementation shall be well-defined.

ALC_TAT.1.2C The documentation of the development tools shall unambiguously define the meaning of all statements used in the implementation.

ALC_TAT.1.3C The documentation of the development tools shall unambiguously define the meaning of all implementation-dependent options.

6.1.17 ATE_COV.2 Analysis of coverage

Developer action elements:

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

Content and presentation of evidence elements:

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

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

6.1.18 ATE_DPT.1 Testing: high-level design

Developer action elements:

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

Content and presentation of evidence elements:

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

6.1.19 ATE_FUN.1 Functional testing

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 of evidence elements:

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

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

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

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

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

6.1.20 ATE_IND.2 Independent testing – sample

Developer action elements:

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

Content and presentation of evidence 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.

6.1.21 AVA_MSU.2 Validation of analysis

Developer action elements:

AVA_MSU.2.1D The developer shall provide guidance documentation.

AVA_MSU.2.2D The developer shall document an analysis of the guidance documentation.

Content and presentation of evidence elements:

AVA_MSU.2.1C The guidance documentation shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation.

AVA_MSU.2.2C The guidance documentation shall be complete, clear, consistent and reasonable.

AVA_MSU.2.3C The guidance documentation shall list all assumptions about the intended environment.

AVA_MSU.2.4C The guidance documentation shall list all requirements for external security measures (including external procedural, physical and personnel controls).

AVA_MSU.2.5C The analysis documentation shall demonstrate that the guidance documentation is complete.

6.1.22 AVA_SOF.1 Strength of TOE security function evaluation

Developer action elements:

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

Content and presentation of evidence elements:

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

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

6.1.23 AVA_VLA.2 Independent vulnerability analysis

Developer action elements:

AVA_VLA.2.1D The developer shall perform a vulnerability analysis.

AVA_VLA.2.2D The developer shall provide vulnerability analysis documentation.

Content and presentation of evidence elements:

AVA_VLA.2.1C The vulnerability analysis documentation shall describe the analysis of the TOE deliverables performed to search for obvious ways in which a user can violate the TSP.

AVA_VLA.2.2C The vulnerability analysis documentation shall describe the disposition of obvious vulnerabilities.

AVA_VLA.2.3C The vulnerability analysis documentation shall show, for all identified vulnerabilities, that the vulnerability cannot be exploited in the intended environment for the TOE.

AVA_VLA.2.4C The vulnerability analysis documentation shall justify that the TOE, with the identified vulnerabilities, is resistant to obvious penetration attacks.

7 TOE Summary Specification

This section presents a functional overview of the TOE; the security functions implemented by the TOE; and the Assurance Measures applied to ensure their correct implementation.

7.1 TOE Security Functions

This section presents the security functions performed by the TOE. The product provides five security functions:

- Security Management
- Identification and Authentication
- Information Flow ControlFunctionality
- Protection of Security Functions
- Audit

Each of the following section will address how the TOE provides these security functions and will indicate the specific Security Functional Requirements that are met by that Security Function.

Table 6 – Mapping of Security Functions to Security Functional Requirements

	FMT_SMR.1	FIA_ATD.1	FIA_UID.2	FIA_UAU.2	FIA_AFL.1	FDP_IFC.1	FDP_IFF.1	FMT_MSA.3	FDP_RIP.1	FPT_RVM.1	FPT_SEP.1	FPT_STM.1	FAU_GEN.1	FAU_SAR.1	FAU_SAR.3	FAU_STG.1	FAU_STG.4	FMT_SMF.1	FMT_MOF.1
Security Management	X	X						X										X	
Identification and Authentication			X	X	X														
Information Flow Control Functionality						X	X		X										
Protection of Security Functions										X	X	X							X
Audit													X	X	X	X	X		

7.1.1 Security Management

The TOE is delivered to the purchaser with an administrator role already defined: Administrator. An account with this role can be used to perform all of the necessary functionality for administering the Firebox® for the Common Criteria evaluation. The Administrator account has complete control of the entire system. Whoever logs into the TOE as an administrator has access to the features and can add to or edit all the settings and policies. The account information for the administrator is stored on the TOE in a database.

The Firebox® provides restrictive default values for information flow. When the Firebox® is powered on for the first time, t has five policies, which are:

- **HTTPS**
  - Listening on 4117, 4103, and 4105 from Private/trusted interface(s) to Device for UI Management
  - Listening on 4100 from Private/trusted interface(s) to Device for Firewall user authentication
- **SSH** - Listening on 4118 from Private/trusted interface(s) to Device
- **Ping** – From Private/trusted interface(s) to Device
- **Host out** – allows any traffic from Device to any
- **Device discovery**[3] – allows any traffic from any to any (out-of-the-box configuration only)

After the Firebox® has been configured to work within the network; the Device discovery rule is no longer needed. The device is row in the normal mode of operation and only has the following four policies active:

- **HTTPS**
  - Listening on 4117, 4103, and 4105 from Private/trusted interface(s) to Device for UI Management
  - Listening on 4100 from Private/trusted interface(s) to Device for Firewall user authentication
- **SSH** - Listening on 4118 from Private/trusted interface(s) to Device
- **Ping** – From Private/trusted interface(s) to Device
- **Host out** – allows any traffic from Device to any

When the Firebox® is brought into the CC mode of operation, the rule set is restricted to only the following two policies:

- **Ping –** from Private/trusted interface(s) to Device
- **Host out –** Any traffic from Device to any

Meets functional requirements: FMT_SMR.1.1, FMT_SMR.1.2, FMT_MSA.3.1, FMT_MSA.3.2, FIA_ATD.1.1, FMT_SMF.1.1

7.1.2 Identification and Authentication

The CLI is accessed via the console port which only allows an administrator with physical access to the Firebox® to authenticate using this method. The TOE verifies the Login ID and Password by comparing it to the stored values for each user. Any attempt to enter commands other than authentication information will result in a failed login attempt, and the administrator will not be permitted to access any TOE functionality or TOE controlled information.

[3] This policy is automatically removed when interface configuration is modified or after the device is discovered; the only way to restore this out-of-the-box policy is to reset the device to factory defaults.

The Firebox® comes prepackaged with one administrator account. This prepackaged administrator account has the ability to create additional administrator accounts. All administrative duties performed on the Firebox® appliance must be performed by an Administrator account.

The Firebox® provides protection against unauthorized users gaining access to the Firebox® by allowing a settable number of unsuccessful login attempts for an account before that account is locked out. The default setting allows for ten failed login attempts. After ten successive unsuccessful attempts have been made, the account is inaccessible. In order to unlock the locked account, an administrator other than the locked out administrator must successfully authenticate to the Firebox® and manually unlock the locked account.

Meets Functional Requirements: FIA_UID.2.1, FIA_UAU.2.1, FIA_AFL.1.1, FIA_AFL.1.2

7.1.3 Information Flow Control Functionality

The Firebox® appliance filters traffic based on policies that are created by authorized administrators. Out-of-the-box, the Firebox® has five[4] policies configured for use. The polices are listed in section 7.1.1.

The Admin account can add additional policies that can be based on the following criteria:

- Source Address of the information
- Destination Address of the information
- What service the traffic is using (HTTP, HTTPS, FTP, etc.)
- The source port of the information
- The destination port of the information
- Interface the traffic arrives or exits on (Internal/External)

The administrator can either allow or deny service through the TOE based on any of the above criteria.

When packets arrive to the Firebox® appliance, there are specific fields that the Firebox® expects. If the packets do not have information in all of the fields, the packets are 'padded' with zeros. This ensures that information from previous packets, including information that has been deleted, is not reused.

Meets Functional Requirements: FDP_IFC.1.1, FDP_IFF.1.1, FDP_IFF.1.2, FDP_IFF.1.6, FDP_RIP.1.1

7.1.4 Protection of Security Functions

The TSP enforcement functions are invoked by the RapidCore engine (RCE) before each function within the TSC is allowed to proceed. As authentication information is transmitted to the Firebox® requesting access, the Login ID and Password are compared to administrators listed in the accounts database on the Firebox® appliance. If there is a match, and the information has been entered correctly, the Firebox® grants the administrator access to perform actions defined by the administrator's role.

All of the Firebox® functionality is provided from within the Firebox® appliance. The Firebox® is a self-contained appliance and as a result it is protected from interference and tampering by untrusted

[4] The HTTPS and SSH policies have been logically separated in this Security Target. On the Firebox® there is one policy that covers both of these policies and is named WGM-From-Trusted.

subjects. Domains of separation are provided at the software, operating system, and hardware layers. The software layer is provided by the Fireware™ Version 8.0, the operating system layer is provided by the linux kernel and kernel processes, and the hardware layer is composed of the physical hardware components within the Firebox® appliance. The only access to the management of the Firebox® appliance in the evaluated configuration is through the CLI. The CLI has a limited functionality that only allows the configuration of the Firebox® appliance and do not provide access to the underlying Linux operating system to execute third party programs.

The operating system clock inside of the Firebox® provides all of the time stamps for the audits. The system clock can only be set by a user assuming an authorized administrator role. Setting the clock is done through the CLI by an authorized administrator. The time stamps are considered reliable because they are all from the same source and only the authorized administrator has access to change the time. Changing the time is also an auditable event, so if the clock has been changed, there will be a record of it.

The TOE restricts the access to all of the functions listed in FMT_MOF.1 to authorized administrators. When an authentication attempt is made, the Login ID and the Password are compared with the accounts database. If the authentication is successful, the role of the administrator is returned, with granted authorization to TOE functionality. If the authentication is unsuccessful, the administrator requesting access to TOE functionality is denied authorization and unable to perform any TOE functionality.

Meets Functional Requirements: FPT_RVM.1.1, FPT_SEP.1.1, FPT_SEP.1.2, FPT_STM.1.1, FMT_MOF.1.1

7.1.5 Audit

The WatchGuard Firebox® X Family audits events in the form of logs. The logs that it can keep are:

- **Event Log** – Keeps records of all the events such as key negotiation activities, denial-of-service attacks, device failures and administrative activities.
- **Traffic Log** – Keeps records of all the traffic going through the appliance and whether or not these streams are passed or blocked according to the current set of policies.
- **Alarm Log** – Records a history of all the alarms that have been triggered by various events or occurrences.

7.1.5.1 Audit Generation

Audits are generated for all administrative events, traffic events, user events and alarm events. The generation of audit events is performed by the syslog daemon (syslogd). Syslogd is a native Linux daemon that is used to generate logs from events performed on a computer or system running Linux. When an auditable event occurs within the system, syslogd is called upon to write the audit to the appropriate audit log.

The following is the information collected in each of the logs:

Table 7 – Audit Log Information

Audit Log Name	Information Collected	Access
Event Log	Date/Time Severity Details	Admin
Alarm Log	Date/Time	Admin

Audit Log Name	Information Collected	Access
	Severity Alarm Name Detail	
Traffic Log	Date/Time Policy Source Address Destination Address Protocol Source Port Destination Port Incoming Interface User Result Extra Information	Admin

7.1.5.2 Audit Review

Audits are stored on the Firebox® and can be viewed using the CLI interface. Reviewing the audit records is an activity only authorize d administrators are authorized to perform.

The Firebox® appliance allows the authorized administrator to search and sort through audit records. The authorized administrator must authenticate to the Firebox® through the CLI before they are able to access the audit records.

7.1.5.3 Audit Storage

Audits are stored within the Firebox®. Log exhaustion behavior of audit records occurs when either the maximum log partition size of four megabytes is reached or the maximum log count entry limit is reached. If either of the two log exhaustion conditions occurs, the Firebox® will shut down all Ethernet interfaces and traffic is stopped from passing through the TOE.

Each type of log (event, alarm, traffic) has two logs: a primary and a rotated (second) log. Each of these logs has a maximum log count. The following table summarizes the limits of each of the Log Types:

Table 8 – Maximum Entries within Each Log

Log Type	File Names	Rotated File Names	Maximum Log Count
Event Logs	Event.log	Eve nt.log.1	1000 entries
Alarm Logs	Alarm.log	Alarm.log.1	1000 entries
Traffic Logs	Traffic.log	Traffic.log.1	5000 entries

The Traffic log has a maximum log count size of 5000 entries in traffic.log and 5000 entries in traffic.log.1 for a combined total of 10,000 entries. Exceeding the maximum log count in the traffic.log file at 5000 entries causes the logging of entries to roll over into the rotated (second) traffic.log.1 file. The Firebox® shuts down all Ethernet interfaces and stops processing traffic at this point, that is, when both the original and the rotated logs are full. It is the responsibility of the Authorized Administrator to export or clear audit records before bringing the Ethernet interfaces back up. If the log partition has exceeded 4 MB, meaning the log partition is full, and the interfaces are brought back up, one traffic entry is recorded, and then the interfaces are disabled.

It is the responsibility of the Authorized Administrator to export or clear the audit records before bringing the Ethernet interfaces back up. Authorized administrators are able to tell that log exhaustion has occurred by viewing the disabled status of the Ethernet interfaces through the console (log exhaustion shuts down all Ethernet interfaces) and viewing the logs themselves.

Meets requirements: FAU_GEN.1.1, FAU_GEN.1.2, FAU_SAR.1.1, FAU_SAR.1.2, FAU_SAR.3.1, FAU_STG.1.1, FAU_STG.1.2, FAU_STG.4.1

7.2 TOE Security Assurance Measures

This section of the ST maps the assurance requirements for a CC EAL 4 level of assurance to the assurance measures used for the development and maintenance of the TOE. Table 9 provides a mapping of the appropriate documentation to the assurance requirements.

The TOE was developed with the following security assurance measures in place, which constitute a CC EAL 4 level of assurance:

- Configuration Management
- Delivery and Operation
- Development
- Guidance Documentation
- Life cycle Support
- Testing
- Vulnerability Assessment

Table 9 – TOE Security Assurance Measures

CC Assurance Requirements	Assurance Measure Title
ACM_AUT.1	WatchGuard Technologies, Inc. Firebox® X Family: Core™ and Peak™ Series with Fireware™ Version 8.0 Configuration Management Plan
ACM_CAP.4	
ACM_SCP.2	
ADO_DEL.2	WatchGuard Technologies, Inc. Firebox® X Family: Core™ and Peak™ Series with Fireware™ Version 8.0 Secure Delivery
ADO_IGS.1	Operating the Firebox® X Family: Core™ and Peak™ Series with Fireware™ Version 8.0 in Common Criteria Mode
ADV_FSP.2	WatchGuard Technologies, Inc. Firebox® X Family: Core™ and Peak™ Series with Fireware™ Version 8.0 Functional Specification
ADV_HLD.2	WatchGuard Technologies, Inc. Firebox® X Family: Core™ and Peak™ Series with Fireware™ Version 8.0 High Level Design

CC Assurance Requirements	Assurance Measure Title
ADV_LLD.1	WatchGuard Technologies, Inc. Firebox® X Family: Core™ and Peak™ Series with Fireware™ Version 8.0 Low Level Design
ADV_RCR.1	WatchGuard Technologies, Inc. Firebox® X Family: Core™ and Peak™ Series with Fireware™ Version 8.0 Representation Correspondence
ADV_SPM.1	WatchGuard Technologies, Inc. Firebox® X Family: Core™ and Peak™ Series with Fireware™ Version 8.0 Security Policy Model
ADV_IMP.1	WatchGuard Technologies, Inc. Firebox® X Family: Core™ and Peak™ Series with Fireware™ Version 8.0 Source Code
AGD_ADM.1	WatchGuard Technologies, Inc. Firebox® X Family: Core™ and Peak™ Series with Fireware™ Version 8.0 User / Administrative Guidance[5]
AGD_USR.1	
ALC_DVS.1	WatchGuard Product Life Cycle version 0.4 Dated 12/1/04 WatchGuard Technologies, Inc. Firebox® X Family: Core™ and Peak™ Series with Fireware™ Version 8.0 Development Environment and Tools
ALC_LCD.1	
ALC_TAT.1	
ATE_COV.2	WatchGuard Technologies, Inc. Firebox® X Family: Core™ and Peak™ Series with Fireware™ Version 8.0 Tests - Functional Tests WatchGuard Technologies, Inc. Firebox® X Family: Core™ and Peak™ Series with Fireware™ Version 8.0 Coverage and Depth Analysis
ATE_DPT.1	
ATE_FUN.1	
ATE_IND.2	
AVA_SOF.1	WatchGuard Technologies, Inc. Firebox® X Family: Core™ and Peak™ Series with Fireware™ Version 8.0 Vulnerability Assessment
AVA_VLA.2	
AVA_MSU.2	

[5] See Table 2 in Section 3 of the Configuration Management document for a list of all user/admin documentation.

See Section 9.7 Rationale for Assurance Requirements for rationale discussing why these assurance measures (in the form of the above-identified documents) are suitable to cover the security assurance requirements for an EAL 4 evaluation.

7.3 Strength of Function Claims

Authentication to the TOE is done through the use of a Login ID and Password session. The administrator must have a personal account that consists of a Name, Password, and a description field that should be used to uniquely identify the user (i.e. the user's real name). The Name must be a minimum of two characters long but not longer than eight characters long, and it must be unique. The account Name is a unique identifier, so there cannot be more than one account for each Name. The Password must be at least eight characters long, but not longer than thirty-two characters long. The strength of function claims apply to the Identification and Authentication Security Function as well as the Security Management Security Function. In addition to SOF claims, the TOE ensures that once a settable amount (3) of authentication attempts is reached (of which can not be zero), the user becomes unable to authenticate to the Firebox® appliance. For the account to be useable again, an authorized administrator of the TOE must unlock the locked account.

8 Protection Profile Claims

The TOE conforms to the U.S. Government Traffic-Filter Firewall Protection Profile for Low-Risk Environments, Version 1.1, April 1999.

8.1 Refinements to Protection Profile

The Security Functional Requirement FIA_AFL.1 has been refined to meet the requirements of this Security Target, but does not follow the requirements set forth in the Protection Profile. FIA_AFL.1 is not used for remote administration as claimed in the Protection Profile, but is used to prevent users from repeatedly attempting to login unsuccessfully at the CLI.

9 Rationale

The rationale provided in this section is taken from the TFFPP.

9.1 Rationale for IT Security Objectives

O.IDAUTH This security objective is necessary to counter the threat: T.NOAUTH because it requires that users be uniquely identified before accessing the TOE.

O.SINUSE This security objective is necessary to counter the threats: T.REPEAT and T.REPLAY because it requires that the TOE prevent the reuse of authentication data so that even if valid authentication data is obtained, it will not be used to mount an attack.

O.MEDIAT This security objective is necessary to counter the threats: T.ASPOOF, T.MEDIAT and T.OLDINF, which have to do with getting impermissible information to flow through the TOE. This security objective requires that all information that passes through the networks is mediated by the TOE and that no residual information is transmitted.

O.SECSTA This security objective ensures that no information is comprised by the TOE upon start-up or recovery and thus counters the threats: T.NOAUTH and T.SELPRO.

O.SELPRO This security objective is necessary to counter the threats: T.SELPRO and T.AUDFUL because it requires that the TOE protect itself from attempts to bypass, deactivate, or tamper with TOE security functions.

O.AUDREC This security objective is necessary to counter the threat: T.AUDACC by requiring a readable audit trail and a means to search and sort the information contained in the audit trail.

O.ACCOUN This security objective is necessary to counter the threat: T.AUDACC because it requires that users are accountable for information flows through the TOE and that authorized administrators are accountable for the use of security functions related to audit.

O.SECFUN This security objective is necessary to counter the threats: T.NOAUTH, T.REPLAY and T.AUDFUL by requiring that the TOE provide functionality that ensures that only the authorized administrator has access to the TOE security functions.

O.LIMEXT This security objective is necessary to counter the threat: T.NOAUTH because it requires that the TOE provide the means for an authorized administrator to control the limit access to TOE security functions.

Table 10 – Mapping of Threats to Security Objectives

	T.NOAUTH	T.REPEAT	T.REPLAY	T.ASPOOF	T.MEDIAT	T.OLDINF	T.AUDACC	T.SELPRO	T.AUDFUL
O.IDAUTH	X								
O.SINUSE		X	X						
O.MEDIAT				X	X	X			
O.SECSTA	X							X	
O.SELPRO	X							X	X
O.AUDREC							X		
O.ACCOUN							X		
O.SECFUN	X		X						X
O.LIMEXT	X								

9.2 Rationale for Security Objectives for the Environment

All of the assumptions stated in section 3.1.1 are considered to be security objectives for the environment. The following are the Security Target non-IT security objectives, which, in addition to those assumptions, are to be satisfied without imposing technical requirements on the TOE. That is, they will not require the implementation of functions in the TOE hardware and/or software. Thus, they will be satisfied largely through application of procedural or administrative measures.

O.PHYSEC The TOE is physically secure.

O.LOWEXP The threat of malicious attacks aimed at discovering exploitable vulnerabilities is considered low.

O.GENPUR There are no general-purpose computing capabilities (e.g. the ability to execute arbitrary code

O.PUBLIC The TOE does not host public data.

O.NOEVIL Authorized administrators are non-hostile and follow all administrator guidance; however, they are capable of error.

O.SINGEN Information can not flow among the internal and external networks unless it passes through the TOE.

O.DIRECT Human users within the physical secure boundary protecting the TOE may attempt to access the TOE from some direct connection (e.g., a console port) if the connection is part of the TOE.

O.GUIDAN This non-IT security objective is necessary to counter the threat: T.TUSAGE because it requires that those responsible for the TOE ensure that it is delivered, installed, administered, and operated in a secure manner.

O.ADMTRA This non-IT security objective is necessary to counter the threat: T.TUSAGE because it ensures that authorized administrators receive the proper training. O.ADMTRA also counters the threat T.AUDACC by helping ensure the audit logs are reviewed.

This security objective for the TOE environment is derived from the assumption stated specifically for this TOE in this ST.

Table 11 – Mappings Between Threats/Assumptions and Security Objectives for the Environment

	T.TUSAGE	T.AUDACC
O.GUIDAN	X	
O.ADMTRA	X	X

Since the rest of the security objectives for the environment are, in part, a re-statement of the security assumptions, those security objectives trace to all aspects of the assumptions.

9.3 Rationale for Not Including All Threats and Objectives from the PP

The following threats and objectives are concerned with remote administration. Remote administration is not included as part of this evaluation and the following threats and objectives has been excluded from this ST.

- T.PROCOM
- O.ENCRYP
- A.NOREMO
- A.REMACC
- O.NOREMO
- O.REMACC

9.4 Rationale for Security Requirements

The rationale for the chosen level of SOF-basic is based on the low attack potential of the threat agents identified in this security target. Those security objectives imply the use of probabilistic or permutational security mechanisms. Because the metrics defined are the minimal "industry" accepted (for the passwords) and government required (for the encryption) metrics, it is rationalized that they are sufficient to provide an SOF-basic level of protection.

FMT_SMR.1 Security roles

Each of the CC class FMT components in this Security Target depend on this component. It requires the administrator to choose a role(s). This component traces back to and aids in meeting the following objective: O.SECFUN.

FIA_AFL.1 Authentication failure handling

This component ensures that users who are not authorized administrators can not endlessly attempt to authenticate. After a settable amount of authentication failures (the default is ten), which cannot be zero, the user becomes unable to authenticate to the Firebox® appliance. This goes on until an authorized administrator makes authentication possible again for that user. This component traces back to and aids in meeting the following objective: O.SELPRO.

FIA_ATD.1 User attribute definition

This component exists to provide users with attributes to distinguish one user from another, for accountability purposes and to associate the role chosen in FMT_SMR.1 with a user. This component traces back to and aids in meeting the following objectives: O.IDAUTH and O.SINUSE.

FIA_UID.2 User identification before any action

This component ensures that before anything occurs on behalf of a user, the users' identity is identified to the TOE. This component traces back to and aids in meeting the following objectives: O.IDAUTH and O.ACCOUN.

FIA_UAU.2 User Authentication Before Any Action

This component ensures that users are authenticated at the TOE. The TOE is permitted to pass information before users are authenticated. Authentication must occur whether the user is a human user or not and whether or not the user is an authorized administrator. If the authorized administrator was not always required to authenticate, there would be no means by which to audit any of their actions. The SOF metric for this requirement is defined in section 5.1.1 to ensure that the authentication mechanism chosen cannot be easily bypassed. This component traces back to and aids in meeting the following objectives: O.IDAUTH and O.SINUSE.

FDP_IFC.1 Subset information flow control

This component identifies the entities involved in the UNAUTHENTICATED SFP (i.e., users sending information to other users and vice versa). This component traces back to and aids in meeting the following objective: O.MEDIAT.

FDP_IFF.1 Simple security attributes

This component identifies the attributes of the users sending and receiving the information in the UNAUTHENTICAED SFP, as well as the attributes for the information itself. Then the policy is defined by saying under what conditions information is permitted to flow. This component traces back to and aids in meeting the following objective: O.MEDIAT.

FMT_MSA.3 Static attribute initialization

This component ensures that there is a default "deny" policy for the information flow control security rules. This component traces back to and aids in meeting the following objectives: O.MEDIAT, O.SECSTA, and O.SECFUN.

FDP_RIP.1 Subset residual information protection

This component ensures that neither information that had flowed through the TOE nor any TOE internal data are used when padding is used by the TOE for information flows. This component traces back to and aids in meeting the following objective: O.MEDIAT.

FPT_RVM.1 Non-bypassability of the TSP

This component ensures that the TSF are always invoked. This component traces back to and aids in meeting the following objective: O.SELPRO.

FPT_SEP.1 TSF domain separation

This component ensures that the TSF have a domain of execution that is separate and that cannot be violated by unauthorized users. This component traces back to and aids in meeting the following objective: O.SELPRO.

FPT_STM.1 Reliable time stamps

FAU_GEN.1 depends on this component. It ensures that the date and time on the TOE is dependable. This is important for the audit trail. This component traces back to and aids in meeting the following objective: O.AUDREC.

FAU_GEN.1 Audit data generation

This component outlines what data must be included in audit records and what events must be audited. This component traces back to and aids in meeting the following objectives: O.AUDREC and O.ACCOUN.

FAU_SAR.1 Audit review

This component ensures that the audit trail is understandable. This component traces back to and aids in meeting the following objective: O.AUDREC.

FAU_SAR.3 Selectable audit review

This component ensures that a variety of searches and sorts can be performed on the audit trail. This component traces back to and aids in meeting the following objective: O.AUDREC.

FAU_STG.1 Protected audit trail storage

This component is chosen to ensure that the audit trail is protected from tampering. Only the authorized administrator is permitted to do anything to the audit trail. This component traces back to and aids in meeting the following objectives: O.SELPRO and O.SECFUN.

FAU_STG.4 Prevention of audit data loss

This component ensures that the authorized administrator will be able to take care of the audit trail if it should become full. Authorized administrators must routinely check audit record logs and view the TOE's Ethernet interfaces, to ensure that traffic is passing. Should an audit data record become full, all Ethernet interfaces are shut down and traffic is unable to pass through the TOE, until an authorized administrator clears the audit trail. This component traces back to and aids in meeting the following objectives: O.SELPRO and O.SECFUN.

FMT_SMF.1 Specification of Management Functions

This component outlines which management functions can be performed on the TOE. This component traces back to and aids in meeting the following objectives: O.SECFUN.

FMT_MOF.1 Management of security functions behavior

This component was chosen and modified to some extent via permitted CC operations in an attempt to consolidate all TOE management/administration/security functions. This component traces back to and aids in meeting the following objectives: O.SECFUN, O.LIMEXT, and O.SECSTA

Table 12 – Summary of Mappings Between TOE Security Functions and IT Security Objectives

	O.IDAUTH	O.SINUSE	O.MEDIAT	O.SECSTA	O.SELPRO	O.AUDREC	O.ACCOUN	O.SECFUN	O.LIMEXT
FMT_SMR.1								X	
FIA_AFL.1					X				
FIA_ATD.1	X	X							
FIA_UID.2	X						X		
FIA_UAU.2	X	X							
FDP_IFC.1			X						
FDP_IFF.1			X						
FMT_MSA.3			X	X				X	
FDP_RIP.1			X						
FPT_RVM.1					X				
FPT_SEP.1					X				
FPT_STM.1						X			
FAU_GEN.1						X	X		
FAU_SAR.1						X			
FAU_SAR.3						X			
FAU_STG.1					X			X	
FAU_STG.4					X			X	
FMT_SMF.1								X	
FMT_MOF.1				X				X	X

9.5 Rational For Refinements of Security Functional Requirements

The following Security Functional Requirements have been refined from the CC with interpretations dated May 2004.

- FMT_SMR.1
- FDP_IFF.1
- FMT_MSA.3
- FAU_GEN.1
- FAU_STG.4

They have been refined to more closely match the functionality of this specific TOE.

- FIA_UAU.2

FIA_UAU.2 has been refined to more clearly define that the TOE processes the login ID and password at the same time before allowing authorization to the administrator requesting access. FIA_UAU.2 is hierarchical to FIA_UAU.1, adding a more appropriate level of refinement to satisfy the Security Functional Requirement addressed here.

9.6 Rationale for Excluding Security Functional Requirements from the PP

The following SFRs have been excluded from this security target:

- FIA_UAU.4
- FCS_COP.1

These requirements deal with remote administration which is not included as part of this evaluation; therefore, these requirements have been omitted from this security target.

Additionally, the following assignment clauses appended to FMT_MOF.1 have been removed:

- Additionally, if the TSF supports remote administration from either an internal or external network:
    - Enable and disable remote administration from internal and external networks;
    - Restrict addresses from which remote administration can be performed;]

The assignments are required if the TOE performs remote administration. Remote Administration is not claimed by the TOE, therefore these requirements have been removed from FMT_MOF.1.

9.7 Rationale for Assurance Requirements

The TOE is intended to be used in a variety of environments, including providing protection for networks from the Internet and other third party networks. The EAL 4 assurance level is consistent with such threat environments, and generally perceived by the consumer as an adequate and necessary level for such security products. U.S. Government Traffic-Filter Firewall Protection Profile for Low-Risk Environments, Version 1.1 provides EAL 2 assurance requirements. In order to claim EAL 4, the evaluation has been augmented from the PP to include all of the EAL 4 assurance requirements stated in Part 3 of the CC.

The chosen assurance level was also selected for conformance with the Firewall family of Protection Profiles and to meet the vendor's customer requirements.

Configuration Management – The Configuration Management documentation provides a description of automation tools used to control the configuration items and how they are used at the WatchGuard and vendor support development facilities. The documentation provides a complete configuration item list and a unique reference for each item. Additionally, the configuration management system is described including procedures that are used by developers to control and track changes that are made to the TOE. The documentation further details the TOE configuration items that are controlled by the configuration management system.

Corresponding CC Assurance Components:

- Configuration Items

Delivery and Operation – The Delivery and Operation documentation provides a description of the secure delivery procedures implemented by WatchGuard to protect against TOE modification during product delivery. The Installation Documentation provided by WatchGuard details the procedures for installing the TOE and placing the TOE in a secure state offering the same protection properties as the master copy of the TOE. The Installation Documentation provides guidance to the administrator on the TOE configuration parameters and how they affect the TSF.

Corresponding CC Assurance Components:

- Delivery Procedures
- Installation, Generation and Start-Up Procedures

Development – The Firebox® design documentation consist of several related design documents that address the components of the TOE at different levels of abstraction. The following design documents address the Development Assurance Requirements:

- The Functional Specification provides a description of the security functions provided by the TOE and a description of the external interfaces to the TSF. The Functional Specification covers the purpose and method of use and a list of effects, exceptions, and errors message for each external TSF interface.
- The High-Level Design provides a top-level design specification that refines the TSF functional specification into the major constituent parts (subsystems) of the TSF. The high-level design identifies the basic structure of the TSF, the major elements, a listing of all interfaces, and the purpose and method of use for each interface.

- The Low-Level Design provides a description of the internal workings of the TSF in terms of modules and their interrelationships and dependencies. The low-level design provides assurance that the TSF subsystems have been correctly and effectively refined.
- The Correspondence Analysis demonstrates the correspondence between each of the TSF representations provided. This mapping is performed to show the functions traced from the ST description to the Low-Level Design.
- The Implementation Representation is in the form of source code, etc. and captures the detailed internal workings of the TSF in support of analysis.

Corresponding CC Assurance Components:

- Functional specification with fully defined external interfaces
- Security enforcing high-level design
- Descriptive low-level design
- Informal correspondence demonstration
- Subset of the implementation of the TSF
- Informal TOE security policy model

Guidance Documentation – The WatchGuard Guidance documentation provides administrator and user guidance on how to securely operate the TOE. The Administrator Guidance provides descriptions of the security functions provided by the TOE. Additionally it provides detailed accurate information on how to administer the TOE in a secure manner and how to effectively use the TSF privileges and protective functions. The User Guidance provided directs users on how to operate the TOE in a secure manner. Additionally, User Guidance explains the user-visible security functions and how they are to be used and explains the user's role in maintaining the TOE's Security. WatchGuard provides single versions of documents which address the Administrator Guidance and User Guidance; there are not separate guidance documents specifically for non-administrator users of the TOE.

Corresponding CC Assurance Components:

- Administrator Guidance
- User Guidance

Life Cycle – This category deals with the aspect of establishing discipline and control in the processes of refinement of the TOE during its development and maintenance. Confidence in the correspondence between the TOE security requirements and the TOE is greater if security analysis and the production of the evidence are done on a regular basis as an integral part of the development and maintenance activities. There are three categories within Life Cycle which are: Development Security, Life Cycle Definition, and Tools and Techniques.

Corresponding CC Assurance Components:

- Development Security
- Life Cycle Definition
- Tools and Techniques

Tests – There are a number of components that make up the Test documentation. The Coverage Analysis demonstrates the testing performed against the functional specification. The Coverage Analysis demonstrates the extent to which the TOE security functions were tested as well as the level of detail to which the TOE was tested. The Depth of Testing document provides analysis to demonstrate that the functional tests provided can sufficiently demonstrate that the TSF operates in accordance with its high-level design. WatchGuard's Test Plans and Test Procedures, which detail the overall efforts of the testing effort and break down the specific steps taken by a tester, are also provided to satisfy the functional testing requirements.

Corresponding CC Assurance Components:

- Analysis of coverage
- Testing: high-level design
- Functional testing

Vulnerability, TOE Strength of Function, and Guidance Misuse Analyses – A Vulnerability Analysis is provided to demonstrate ways in which an entity could violate the TSP and provide a list of identified vulnerabilities. Additionally, the document provides evidence of how the TOE is resistant to obvious attacks. The Guidance Misuse Analysis aims to ensure that misleading, unreasonable and conflicting guidance is absent from the guidance documentation and secure procedures for all modes of operation have been addressed. The Strength of TOE Security Function Analysis demonstrates the strength of the probabilistic or permutational mechanisms employed to provide security functions within the TOE and how they exceed the minimum SOF requirements.

Corresponding CC Assurance Components:

- Strength of TOE Security Function evaluation
- Examination of guidance
- Developer vulnerability analysis

9.8 Rationale for Not Satisfying All Dependencies

Functional component FMT_MSA.3 depends on functional component FMT_MSA.1 Management of security attributes. In an effort to place all the management requirements in a central place, FMT_MOF.1 was used. FMT_MOF.1 restricts the security attributes and security functions behavior to authorized administrators only. As done previously, the convergence of FMT_MSA.1 and FMT_MSA.3 under FMT_MOF.1 simplifies the documentation and provides all needed functionality under one section. FMT_MOF.1 addresses both security attributes and security functionality, so the need to have FMT_MSA.1 is not needed. Therefore FMT_MOF.1 more than adequately satisfies the concerns of leaving FMT_MSA.1 out of this Security Target.

Functional component FCS_COP.1 deals with remote administration and therefore the rationale for not satisfying all dependencies included in the PP is irrelevant.