FireEye MAS and MPS 2000, 4000, and 7000 Series with Central Management System v.5.0 Security Target

Version 1.2
August 30, 2010

Prepared for:
FireEye, Inc.
1390 McCarthy Blvd.
Milpitas, CA 95035

Prepared by:
Booz Allen Hamilton
Common Criteria Testing Laboratory
900 Elkridge Landing Road, Suite 100
Linthicum, MD 21090-2950

Table of Contents

List of Figures

List of Tables

1 Security Target Introduction

This chapter presents the Security Target (ST) identification information and an overview. An ST contains the Information Technology (IT) security requirements of an identified Target of Evaluation (TOE) and specifies the functional and assurance security measures offered by the TOE.

1.1 ST Reference

This section provides information needed to identify and control this ST and its Target of Evaluation. This ST targets Evaluation Assurance Level 2 (EAL2) augmented with ALC_FLR.2.

1.1.1 ST Identification

ST Title:	FireEye MAS and MPS 2000, 4000, and 7000 Series with Central Management System v.5.0
ST Version:	1.2
ST Publication Date:	August 30, 2010
ST Author:	Booz Allen Hamilton

1.1.2 Document Organization

Chapter 1 of this ST provides identifying information for the FireEye MAS and MPS 2000, 4000, and 7000 Series with Central Management System v.5.0. It includes an ST Introduction, ST Reference, ST Identification, TOE Reference, TOE Overview, and TOE Type.

Chapter 2 describes the TOE Description, which includes the physical and logical boundaries, and describes the components and/or applications that are excluded from the evaluated configuration.

Chapter 3 describes the conformance claims made by this ST.

Chapter 4 describes the Security Problem Definition as it relates to threats, Operational Security Policies, and Assumptions met by the TOE.

Chapter 5 identifies the Security Objectives of the TOE and of the Operational Environment.

Chapter 6 describes the Extended Security Functional Requirements (SFRs) and Security Assurance Requirements (SARs).

Chapter 7 describes the Security Functional Requirements.

Chapter 8 describes the Security Assurance Requirements.

Chapter 9 is the TOE Summary Specification (TSS), a description of the functions provided by FireEye MAS and MPS 2000, 4000, and 7000 Series with Central Management System v.5.0 to satisfy the SFRs and SARs.

Chapter 10 is the Security Problem Definition Rationale and provides a rationale or pointers to a rationale, for security objectives, assumptions, threats, requirements, dependencies, and PP claims for the chosen EAL, any deviations from CC Part 2 concerning SFR dependencies, and a mapping of threats to assumptions, objectives, and SFRs. It also identifies the items used to satisfy the Security Assurance Requirements for the evaluation.

1.1.3 Terminology

This section defines the terminology used throughout this ST. The terminology used throughout this ST is defined in Table 1-2: Terminology Definitions. This table is to be used by the reader as a quick reference guide for terminology definitions.

Terminology	Definition
Administrator	User of the TOE who has access to both administrative functions and monitor functions.
Analyzer	IDS component that performs analysis functions on suspicious data or traffic to determine threats.
Attack	A botnet or malware callback event on the system.
Attacker	An entity that attempts to send malicious code or traffic to a system on the installed network.
Botnet	Set of software "robots" or "zombies" that are controlled remotely by a command and control server.
Botnet server	Command and control server that directs the operation of a botnet.
Callback event	Callback events are generated when the appliance observes outbound communications associated with a remote Command and Control server (C&C). This could include botnet command and control communications, uploads of confidential information as well as downloads of secondary payloads (such as keyloggers or spyware). Callback events indicate that there is an established communication between a bot-infected host and its C&C Server.
Command-line interface	The FireEye appliance has a CLI interface for administering the appliance.
Central Management System	Has a web-based graphical user interface for managing multiple FireEye appliances.
Event	Indicates a type of security intrusion or attack.
Guest image	Software image for an operating system and applications that is run in a virtual machine to analyze suspicious or captured traffic.
Graphical User Interface	The FireEye appliance has a web-based GUI for managing the appliance.
Heuristic analysis	Expert-based analysis that determines the susceptibility of a system towards particular threats using various decision rules or weighing methods.
Infection	When a machine on the network has malware or botnet programs.
Malware	Malicious software used by attackers to disrupt, cause data loss, or gain unauthorized access to computer systems.
MAX Network	A multi-enterprise alliance focused on protecting customers from botnets and other stealthy, targeted malware. The ability to connect to the MAX Network to receive signature updates and to upload detected malware is

	included in the evaluated configuration. The MAX Network itself is a component of the operational environment in the evaluated configuration because it is a server that sits in a server room at FireEye HQ. It's a trusted IT product with which the TOE can interact, but it's not considered part of the TOE since it belongs to the vendor and not the customer.
Monitor	User of the TOE who only has access to monitoring functions.
Role	Assigned to a user, allows users controlled access to TOE components. In this case, the three roles are Administrator, Monitor, and LCD panel administrator.
rsyslog	An open source program for forwarding log messages in an IP network for UNIX and Unix-like systems.
Sandbox	A closed environment in which malware is submitted and its effects on virtual machines are reported.
Scanner	IDS component that actively looks through data flows and traffic to find suspicious items.
Sensor	IDS component that views data flows and traffic passing through to find suspicious items.
System Administrator	See Authorized System Administrator.
User	In the evaluated configuration, a user is a global term for Administrators and Monitors.
Virtual Machine	A software program that runs an instance of an operating system. The operating system runs on top of a program that emulates a hardware system. In the evaluated configuration, each VM is isolated by address space and their virtual connections are isolated by bridges.
Zero-day attack	An attack by malware that exploits unknown or newly discovered vulnerabilities in software before they become known or before security patches are applied to fix them.

Table 1-1: Customer Specific Terminology

Term	Definition
External IT entity	Any IT product or system, trusted or not, outside of the TOE that interacts with the TOE.
Role	A predefined set of rules establishing the allowed interactions between an end user and the TOE.
TOE Security Functions (TSF)	A set consisting of all hardware, software, and firmware of the TOE that must be relied upon for the correct enforcement of the TSP.
User	Any entity (human user or external IT entity) outside the TOE that interacts with the TOE.

Table 1-2: CC Specific Terminology

1.1.4 Acronyms

The acronyms used throughout this ST are defined in Table 1-3: Acronym Definitions. This table is to be used by the reader as a quick reference guide for acronym definitions.

Acronym	Definition
CC	Common Criteria
CCEVS	Common Criteria Evaluation and Validation Scheme
CCIMB	Common Criteria Interpretations Management

	Board
CLI	Command-line Interface
CMS	Central Management System
COTS	Commercial Off the Shelf
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
GUI	Graphical User Interface
EAL	Evaluation Assurance Level
FTP	File Transfer Protocol
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
IDS	Intrusion Detection System
IP	Internet Protocol
IRC	Internet Relay Chat
IT	Information Technology
LCD	Liquid Crystal Display
MAS	Malware Analysis System
MAX	Malware Analysis and Exchange
MPS	Malware Protection System
NIAP	National Information Assurance Partnership
OS	Operating System
OSI	Open System Interconnection
PCM	Platform Configuration and Management
PP	Protection Profile
SNMP	Simple Network Management Protocol
SMTP	Same Message Transfer Protocol
SSH	Secure Shell
SSL	Secure Sockets Layer
ST	Security Target
TCP	Transmission Control Protocol
TFTP	Trivial File Transfer Protocol
TOE	Target of Evaluation
TSF	TOE Security Function
UI	User Interface
URL	Uniform Resource Locator
VM	Virtual Machine

Table 1-3: Acronym Definitions

1.1.5 References

[1] Common Criteria for Information Technology Security Evaluation, CCMB-2009-07-004, Version 3.1 Revision 3, July 2009

[2] U.S. Government Protection Profile Intrusion Detection System System For Basic Robustness Environments Version 1.7

[3] FireEye CMS Operator's Guide Version 5.0

[4] FireEye 4200 Appliance v4.2.0 Evaluator's Guide

[5] FireEye CLI Command Reference Guide Version 5.0

[6] FireEye Appliance Operator's Guide Version 5.0

[7] FireEye CMS Quick Start

[8] FireEye Appliance Quick Start

[9] FireEye Appliance Release Notes Version 5.0

[10] Product Requirements Document Release 5.0.

1.1.6 CC Concepts

The following are CC concepts as used in this document. A Subject is any user of the TOE (account, user, administrative user). An Object (i.e., resource or entity) can be a dataset, volume, command issued by a user, etc. An Operation is any action on a resource (e.g. read, write, create, fetch, update, control, alter, or scratch). A Security Attribute is information such as username, groups, profiles, facilities, passwords, etc. that is kept in the security file for the user. An External Entity is anything outside of the TOE that affects the TOE.

1.1.7 Protection Profile

This ST claims conformance to the U.S. Government Protection Profile Intrusion Detection System System for Basic Robustness Environments Version 1.7 (herein referred to as the IDS System PP). The IDS System PP states the following:

> *This PP makes a distinction between the System and TOE. The term System is used when the PP is referring to the ID monitoring, analysis, and reaction mechanisms as specified by the IDS security function requirements Class. When the term TOE is used, the PP is referring to the complete IT product that implements all TOE Security Function Requirements necessary to ensure accountability and protection for the ID monitoring, analysis, and reaction capabilities.*

Using the definition of "system" in the IDS System PP, the "authorized System administrator" in the PP has the ability to modify the IDS components of the TOE. "Administrators" in this ST have the ability to modify the IDS components of the TOE and therefore maps to the term "authorized System administrator" in the IDS System PP. Likewise, the "authorized administrator" in the IDS PP requires access to the IDS (system) data. "Monitor" in this ST has access to the IDS (system) data and therefore maps to the term "authorized administrator" in the IDS System PP.

1.2 TOE Reference

1.2.1 TOE Identification

FireEye MAS and MPS 2000, 4000, and 7000 Series with Central Management System v.5.0

1.3 TOE Overview

FireEye MAS and MPS 2000, 4000, and 7000 Series with Central Management System v.5.0, (herein referred to as FireEye AKA the TOE), detects malware by analyzing suspicious network flows in virtual victim machines. The FireEye appliance identifies malicious attacks, including those targeting web browsers. It secures against both widespread and targeted network malware without relying on manual IT analysis. Signature matching is used in the IDS process, but the IDS process does not rely on the signature matching components or updated signatures to function properly. After definitively confirming a targeted malware attack, the FireEye appliance is integrated into a network to block the attack, quarantine the infected host and alert Administrators to the incident.

FireEye appliances come in different forms, or series. Each FireEye appliance has the same basic functionality and hardware components, including but not limited to: a management interface, a live malware analysis port, traffic monitoring ports, a VGA port, a serial port, and two USB ports. Additionally, each appliance can support a number of simultaneous Virtual Machines, and can offer a specific standard of throughput.

The breakdown of each of the series is shown in the following table:

	Traffic Monitoring Ports	Physical Appliance Size	LCD Panel?	Throughput
2000 Series	2	1U half-depth	No	20-45 Mbps
4000 Series	4	1U full-depth	Yes	250 Mbps
7000 Series	4	2U full-depth	Yes	1 Gbps
Central Management System (CMS)	N/A	1U full-depth	Yes	N/A

Table 1-4: FireEye Series Comparison

The TOE:

- Provides accurate Web malware protection using virtual victim machine analysis
- Utilizes multi-stage detection to eliminate false alerts and need for complex software agents
- Performs continuous network traffic analysis in an out-of-band deployment
- Includes automatic creation of dynamic signatures based on malware executable code

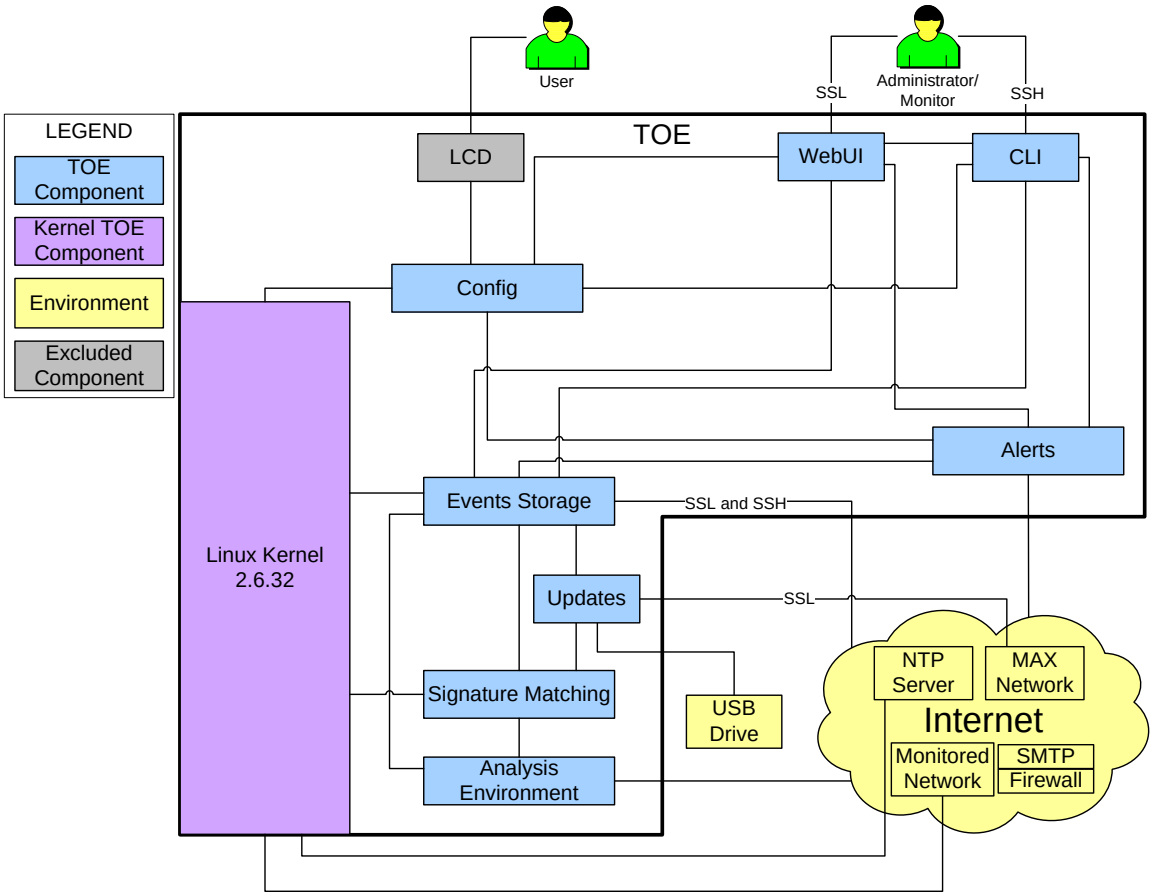

Figure 1 – TOE Boundary for MPS and Malware-Analysis Appliances

As illustrated in Figure 1, the TOE boundary contains 9 components – the Kernel, Config, CLI, WebUI, Events Storage, Signature Matching, Analysis Environment, Updates, and Alerts. The Kernel is Linux v2.6.32 and performs basic system functionality that is important in the TOE. The Kernel is physically contained in the FireEye appliance and provides OS functionality to the TOE, including capture, clock, and audit functionalities. It also provides a file system and memory management. The basic functionality of the operating system beyond this is not provided to TOE users via the remote interfaces that are used in the evaluation.

When network traffic enters a FireEye appliance, it is captured through the Kernel. The Kernel begins the analysis process by determining if the traffic is suspicious. If it is determined to be suspicious, it is then put through the Signature Matching component. The specific processes of the Signature Matching component the traffic is sent through are dictated by the license of the appliance. Since the TOE includes three separate appliances, all of the relevant processes are executed in the evaluated configuration. The Signature Matching component runs analysis tests based on signatures and historical patterns to determine if the traffic is a known threat. If the traffic matches a signature or pattern that FireEye recognizes, the traffic is sent directly to the Events Storage

component for storage, and then the Alerts component sends alerts to selected users. If the suspicious traffic does not match a known signature or pattern, does not match certain heuristics, or exceeds certain statistical thresholds, it is then sent to the Analysis Environment, which utilizes a Virtual Machine (VM) and Virtual Network (VN). The traffic is run through the VM/VN and monitored to determine if any malicious behaviors are detected. If a threat is detected, the Analysis Environment component sends it to the Events Storage component where it is time stamped and stored for future reference. The threat is also sent to the Alerts component, which first checks with the Config component to determine which alarm method to use. Once this is determined, the Alerts component sends an alarm to the selected users.

There are two roles maintained by the TOE – Administrators and Monitors. Likewise, there are two types of user interfaces to the TOE inside the TSF: WebUI and Command-line Interface (CLI). Refer to Table 1-4 for more information on the interfaces included for each appliance.

Administrators have full access to the monitoring and administrative functions, and can access the WebUI and CLI. Administrative functions in the WebUI allow Administrators to edit aspects of the system, including but not limited to: editing user accounts, network settings, and overall TOE settings. Monitors only have access to monitoring functions, and do not have full access to the TOE's interfaces, specifically the CLI administrative functions and the Appliance Settings tabs located inside the WebUI. For more information on management functions capable of being performed on the TOE, see Section 7.2.2.

The primary way that both Administrators and Monitors can view or change information is the WebUI. The WebUI is a browser-based component that allows users to perform all the functions the TOE allows them to, based on role. A user points a web browser to the IP of the WebUI, and then must identify and authenticate to the TOE's Apache server v2.2.3 via username and password in order to connect to the TOE. To access the WebUI, a user must use a supported browser: Firefox 3.0.5 or greater, or Internet Explorer 7.0 or greater.

Along with the WebUI, Administrators have access to the CLI and the administrative functions available using this interface, where they can perform the same functions as in the WebUI but in a text-based environment. The CLI is a shell program that runs on top of the Linux OS. It replaces standard Linux shells for the users of the appliance.

Any changes made to the TOE configuration in any of the specified user interfaces are transferred and stored in the Config component of the TOE.

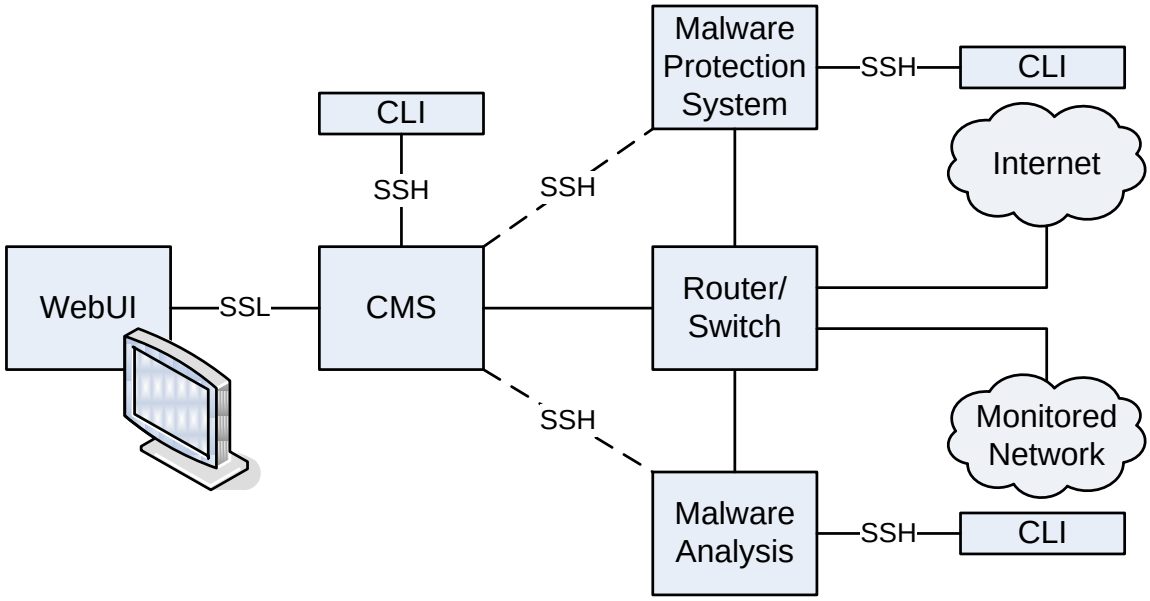

Figure 2 – TOE Deployment (SPAN/mirror MPS)

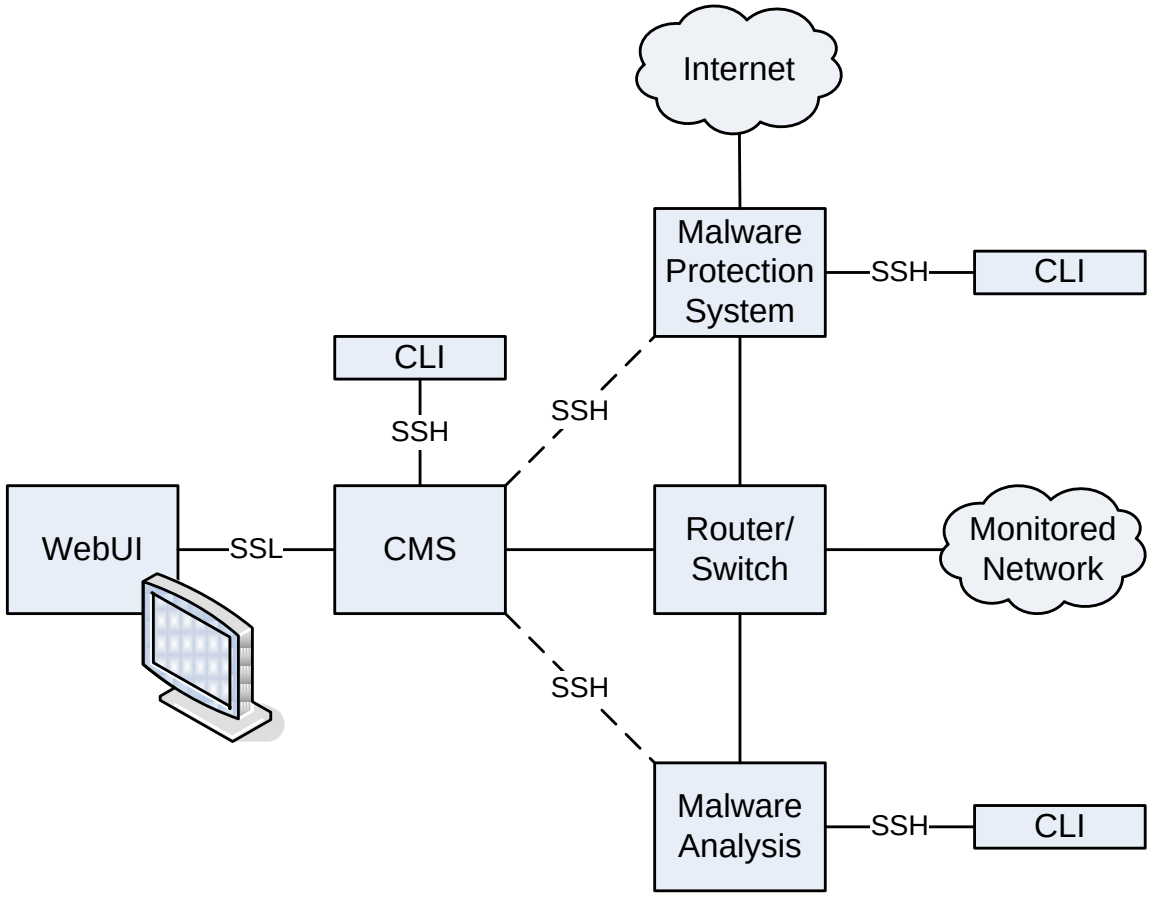

Figure 3 – TOE Deployment (Inline MPS)

As illustrated in Figures 2 and 3, the TOE deployment contains three instances of FireEye: Central Management System (CMS), Malware Protection System (MPS) and Malware-Analysis appliances. Each of the FireEye Series operate on the same TOE boundary as shown in Figure 1. However, as described in Table 1-5, the functionality of each instance is different. The MPS and Malware-Analysis FireEye instances have the WebUI interface, but that functionality is delegated to the Central Management System (CMS) for the evaluated configuration.

The difference between Figures 2 and 3 is whether the MPS is inline or not. The evaluated configuration will test both of these functionalities.

Each appliance needs to be connected to the CMS manually. Once connected, the CMS polls for data from each of the appliances every 5 minutes. Connection to each appliance is done via the standard SSH based authentication mechanisms. The CMS periodically retries connection to each appliance upon failure.

Note: The dotted line connections in the deployment diagrams do not represent a physical connection. They represent the logical connection of the MPS and Malware-Analysis appliances opening an SSH connection to the CMS.

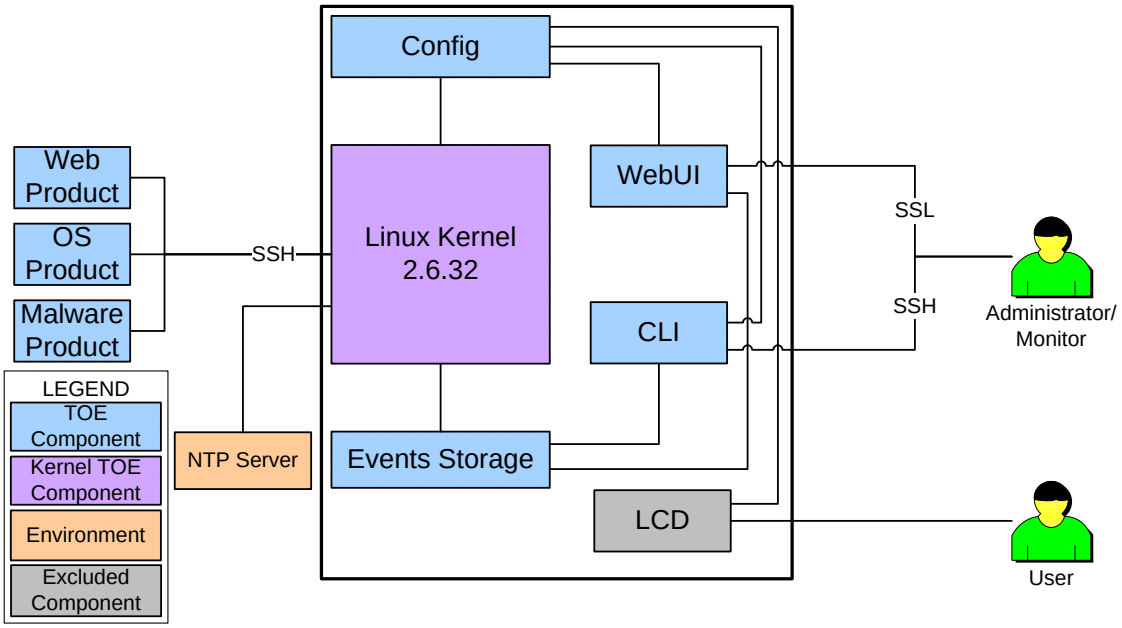

Figure 4 – Central Management System TOE Boundary

Figure 4 shows the TOE boundary for the Central Management System. The CMS consists of the Kernel, Config, CLI, WebUI, and Events Storage components. It collects data from all the other FireEye appliances in the deployment and aggregates the data. Therefore, the CMS does not perform any of the IDS functionality of the TOE. Its main purpose is to allow for a single point for user interaction with multiple appliances in the TOE.

Table 1-5 details the differences between the MPS, Malware-Analysis, and CMS appliances.

FireEye Appliance	Description
FireEye Malware Protection System Appliance	The internal virtual victim machines of the FireEye Malware Protection System (MPS) appliance have two functions. First, it emulates the browser (client) side of suspicious web transactions between actual network users and web servers to determine if the web server is attempting to infect the browser. Suspicious code is replayed into and analyzed inside the appliance's virtual machines (VMs), enabling it to discover polymorphic or zero-day malware that may not have been seen before. Second, the internal victim machines of the FireEye MPS appliance run vulnerable versions of Windows Services that might be the targets of direct network attacks. As in the first function, suspicious transactions between actual network hosts are replayed and the appliance can detect malicious behavior that is polymorphic or novel. This appliance detects known Malware Callback events and attacks that use either OS exploits or web infections to propagate. This appliance can also be run in-line on the monitored network as sort of an intrusion prevention system. Instead of using SPAN/mirror ports, this appliance takes in all incoming data and will drop packets that it deems to be malicious, and forward the rest.
FireEye Malware Analysis Appliance	If a malware file or location (such as a URL) has already been identified by the FireEye appliance or another method, the FireEye Malware Analysis appliance can be used without a live monitored network connection. For each file or URL, the appliance generates a report that lists the characteristics of the malware and its effects. Note: All FireEye appliances use the ether1 interface for management, not analysis. FireEye Malware Analysis can act on any of the following inputs: • A URL pointing to a file located on the internet. HTTP, HTTPS and FTP URLs are supported. Such a URL is referred to as malicious-file-URL. • A file located on a desktop. • A list file containing URLs that point to files on the Internet. This product accepts malicious files or URLs submitted by the user and analyzes them in the FireEye Virtual Victim Environment. The analysis provides detailed information of the OS and Network behavior of the malicious entity. Each malicious entity is analyzed in a protected sandbox environment. In a Sandbox environment, the analysis is conducted in a way that no network communication generated from these analyses leaves the appliance. Note: After being used for analysis, the Virtual Machine is destroyed. New VMs are loaded off clean snapshots so that the effect of malware can be isolated to that particular suspected traffic.
FireEye Central Management System (CMS)	The FireEye CMS appliance provides the infrastructure, features, and support to manage multiple FireEye appliances and to aggregate detection and protection data across the managed appliances: • Central event collection—View events and attacker information from multiple appliances and apply user-configurable filters with fast database queries.

	• Central management and control—Group appliances for effective user management, configuration changes, updates, and alert settings. • Host-centric user interface—View status based on hosts, rather than events, for better management and the ability to take more effective action. The CMS' administrative functions allow it to determine what is audited in the event logs of each other FireEye device in the system. Additionally, the CMS is the main point for data viewing, as it can aggregate all of the collected event and alert logs from the MPS and Malware-Analysis aspects of the evaluated configuration. The CMS can either poll the MPS and Malware-Analysis instances for their System data. Data from other appliances passes through an aggregation component and is passed to a communications component, which encrypts the data to be sent to the CMS.

Table 1-5: TOE Deployment

In order to have a complete IDS system for the evaluated configuration, the MPS and Malware-Analysis appliances must both be purchased together. Therefore, both appliances are included in the evaluated configuration. Internally, they are all identical and are represented by the TOE Boundary diagram in Figure 1. The information they capture and the internal interfaces and modules used are different and is determined by the licensing.

The CMS is a stripped-down version of the TOE hardware and software that represents a common WebUI to view aggregate data across multiple TOE appliances. It does this by establishing SSH trusted channels to all other devices and storing their data in its own internal database. The internals of the CMS are limited to a CLI for self-configuration, a WebUI that is identical to what is outlined in this ST for the TOE appliances, Platform Configuration and Management (PCM) for its own internal configuration, and an internal database to aggregate the information from other devices.

1.4 TOE Type

The TOE type for FireEye is an Intrusion Detection System (IDS). CCEVS defines IDS as the following: "Devices generally deployed on networks or user hosts to monitor traffic and look for evidence of unauthorized intrusions or network attacks." According to the IDS PP, "An Intrusion Detection System (IDS) monitors an IT System for activity that may inappropriately affect the IT System's assets. An IT System may range from a computer system to a computer network. An IDS System (System) consists of Sensors, Scanners and Analyzers (i.e., IDS components). Sensors and Scanners collect information regarding IT System activity and vulnerabilities, and they forward the collected information to Analyzers. Analyzers perform intrusion analysis and reporting of the collected information."

These components pertain to the TOE in the following manner:

- The TOE functions as a sensor by passively receiving traffic over a SPAN port or network tap.

- The TOE functions as a scanner by providing administrators the ability to actively upload suspected malware for analysis.
- The TOE functions as an analyzer by examining traffic both against known malware and botnet signatures and by performing VM analysis against suspected zero-day traffic.

2 TOE Description

This ST claims conformance to the IDS System PP v1.7. The IDS System PP specifies the minimum security requirements for a TOE that is a System. A System is one or more Sensors and/or Scanners, and one or more Analyzers. A System monitors an IT System for activity that may inappropriately affect the IT System's assets, performs analysis on the data it collects, and reacts appropriately. The information collected may be obtained from a variety of sources located on an IT System. Similarly, the response functions may affect one or more targets on the IT System.

The IDS System PP makes a distinction between the System and TOE. The term System is used when the PP is referring to the ID monitoring, analysis, and reaction mechanisms as specified by the IDS security function requirements class. When the term TOE is used, the PP is referring to the complete IT product that implements all TOE Security Function Requirements necessary to ensure accountability and protection for the ID monitoring, analysis, and reaction capabilities.

2.1 Evaluated Components of the TOE

The following table describes the TOE components in the evaluated configuration:

Component	Definition
Config	Component of FireEye that contains and modifies all FireEye system configurations, user configurations and auditing options.
CLI	Command-line interface that uses OpenSSL SSH functionality and allows Administrators to perform administrative functions. Monitors do not have access to the CLI.
WebUI	Browser-based interface that uses OpenSSL and allows Administrators to perform administrative functions, and allows both Administrators and Monitors to perform monitoring functions.
Linux kernel v2.6.32	The Kernel is Linux v2.6.32 and holds basic system functionality that is important in the TOE. The kernel is physically contained in the FireEye appliance and provides OS functionality to the rest of the TOE, including capture, clock, and audit functionalities. The basic functionality of the operating system beyond this is not security relevant for the evaluated configuration.
Analysis Environment	Creates and manages virtual machines that are used for simulated traffic to determine if suspicious traffic and binaries are malicious in nature.
Signature Matching	Checks data against known malware and botnet traffic to determine if the traffic needs to be run by the analysis environment.
Events Storage	Records information regarding infections and callbacks on systems within the network, and applies basic identifying information.
Alerts	Mechanism for notifying Administrators or Monitors in the event of a detected infection or callback.

Table 2-1: Evaluated Components of the TOE

2.2 Components and Applications in the Operational Environment

The following table lists components and applications in the environment that the TOE relies upon in order to function properly:

Component	Definition
Internet	The Internet contains the Monitored Network, the NTP Server, and the MAX Network. Additionally, the TOE can configure alerts to be sent to the Internet via SMTP, SNMP, and HTTP POST methods. The command line SMTP client used for email notifications is v2.5.1.
Monitored Network	In the evaluated configuration, all internet traffic passing through the switch FireEye is connected to is also passed into FireEye. This data gets sent through the Statistical, Signature, and Heuristic analysis. If the traffic is determined to be suspicious from using any of the previous analysis methods, then the traffic is sent through Virtual Machine analysis All data transferred, including but not limited to URLs, executables, and code, is evaluated.
NTP Server	FireEye appliances utilize NTP Servers by default. An NTP Server keeps the system up to date with the latest system time from their servers. In this case, it is used for accurate timestamps on audit and system data.
FireEye Malware Analysis and Exchange (MAX) Network	The FireEye MAX Network circulates the latest malware analysis intelligence to participating FireEye appliances, ensuring customer data, intellectual property, and resources are safeguarded from the threat of network malware and botnets. The ability to connect to the MAX Network to receive signature updates and to upload detected malware is included in the evaluated configuration. The MAX Network itself is a component of the operational environment in the evaluated configuration because it is a server that sits in a server room at FireEye HQ. It's a trusted IT product with which the TOE can interact, but it's not considered part of the TOE since it belongs to the vendor and not the customer.
USB	While system updates can come from the Internet, a user can also load the updates onto a USB drive and plug it into the FireEye appliance physically. This allows users an alternate way to install updates, which must be encrypted on upload and decrypted on install. USB drives also cannot be mounted to install untrusted software to FireEye.

Table 2-2: Evaluated Components of the Operational Environment

2.3 Excluded from the TOE

The following optional products, components, and/or applications can be integrated with FireEye but are not included in the evaluated configuration. They provide no added security related functionality. They are separated into three categories: not installed, installed but requires a separate license, and installed but not part of the TSF.

2.3.1 Not Installed

Since the TOE is a piece of COTS hardware, there are no software components that are prerequisites for TOE installation and operation.

2.3.2 Installed but Requires a Separate License

There are no components that are installed with FireEye but require a separate license and are therefore not included in the TOE boundary.

2.3.3 Installed But Not Part of the TSF

These components are installed with FireEye appliances, but are not included in the TSF.

- LCD Front Panel – The password protected LCD front panel that is physically located on the FireEye appliances (except for the 2000 series) allows users to perform limited TOE configuration. Since a username is not required to authenticate to the LCD panel, it is assumed that individuals with physical access to the TOE will also be users of the TOE. The LCD panel is intended for initial configuration, and therefore is not considered part of the TSF for the evaluated configuration. Additionally, initial setup can be performed using the serial port and a machine that can connect using SSH. This method is detailed in the user guides.

2.4 Physical Boundary

2.4.1 Hardware

FireEye is a rack-mounted device. It contains the following components on the front panel:

- Network ports—Port 1 is used as management interfaces, while port 2 is used by the FireEye Malware Analysis appliance. This port is disabled for other appliances. The other 2 or4 ports are used for monitoring network traffic.
- DB-9 serial port—used for serial console connection.
- VGA port—used to connect a monitor.
- LCD panel—Used to read status and configuration information. The LCD panel is not included in the 2000 Series. The LCD panel usage is not included in the evaluated configuration.
- USB ports— The USB key is first formatted so that there is assurance that no residual data is present to potentially infect the disconnected host. Used for

upload/download of configuration and signature data and updates from the MAX Network.

- Configuration buttons—Used to configure basic network settings when connecting the appliance to the network. These buttons also allow users to view basic network configurations.

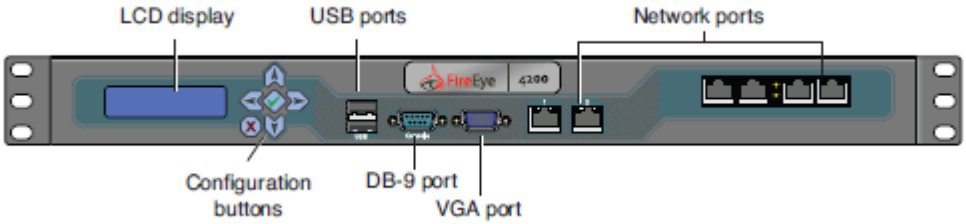

Figure 5 – FireEye Front Panel Components

Additionally, the CMS appliance has the following physical interfaces:

- 2 management interfaces (RJ-45)
- 1x serial port
- 1x LCD

The below hardware components are within the Operational Environment and are not provided as part of TOE. Therefore, these components are in the Physical Boundary of the Operational Environment. Refer to Table 2-2 for more information.

- Internet
- Monitored Network
- NTP Server
- FireEye Malware Analysis and Exchange (MAX) Network
- USB device

Note: Some of these Operational Environment components are both hardware and software based.

2.4.2 Software

The Operating System that FireEye appliances run on is the CentOS 5.3 distribution of Linux. CentOS has more security features than normal Linux and has a greater focus on minimalism. It includes shell and web server components that are required for the overall operations of the box. However, it does not include any package management or updating components. It also does not include any development components. Additional restrictions have been added to limit access to the box and to the base Linux environment.

The web server used by the TOE is Apache 2.2.3, which uses Ruby on Rails.

The TOE for this Security Target is the FireEye MAS and MPS 2000, 4000, and 7000 Series with Central Management System v.5.0. The below software components are installed on the FireEye appliance:

- Linux kernel v2.6.32
- Config
- CLI
- WebUI
- Events Storage
- Signature Matching
- Analysis Environment
- Alerts

2.5 Logical Boundary

The logical boundary of the TOE is described in the terms of the security functionalities that the TOE provides to the systems that utilize this product for intrusion detection.

The logical boundary of the TOE will be broken down into six security classes: Security Audit, Identification and Authentication, Security Management, Protection of the TSF, Encrypted Communications, and Intrusion Detection System. Listed below are the security functions with a listing of the capabilities associated with them:

2.5.1 Security Audit

When auditable events occur, such as access to the TOE or access to system data, the TOE generates audit logs for all TOE and user actions that are time stamped and stored in the appropriate internal database(s). Only Administrators have the ability to view these records through the Command Line Interface (CLI) and sort these records based on predetermined criteria. The TOE allows events that are deemed auditable to be either excluded or included from the set of auditable events based on event type. All audit records are protected from unauthorized deletion and modification. When the audit storage location has reached its maximum capacity, all audit records with the exception of the oldest log files, will be maintained. Alarms are sent to the appropriate individuals when old audit records are overwritten as a result of the audit storage meeting its capacity. For more detailed information on security audit, see section 9.1.1.

2.5.2 Identification & Authentication

All users must be identified and authenticated to the TOE via username and password before being allowed to perform any actions on the TOE. The exception to this is that users are allowed to perform TOE functions via the password protected LCD panel without identifying themselves to the TOE. Since a username is not required to authenticate to the LCD panel, it is assumed that individuals with physical access to the

TOE will also be users of the TOE. The LCD panel is meant for initial setup only, and as such is not part of the TSF for the evaluated configuration. The TOE maintains specific security attributes about users in order to correctly identify them with their TOE-associated abilities as well as for future authentication attempts. If a user enters incorrect credentials multiple times, he or she is forbidden from re-attempting to authenticate until a set amount of time has elapsed. The number of incorrect attempts allowed is pre-determined by the Administrator. In addition, the TOE appliances authenticate to the MAX Network and the MAX Network authenticates to the TOE appliances in order to pass updates to the Updates component. This authentication is performed through the use of vendor supplied username and password and through the use of certificates. For more detailed information on identification and authentication, see section 9.1.2.

2.5.3 Security Management

The TOE maintains two roles – Administrator and Monitor. Users under the Administrator role have the ability to perform all administrative functions (e.g. user management, audit management) and monitoring functions.

Users under the Monitor role are able to perform all changes pertaining to monitoring functionality, but are not allowed to perform any other administrative functionality (i.e. user management, audit configuration). Users can perform limited configuration functions via the LCD panel. All functions performed from the LCD panel can also be performed from the WebUI or CLI once the user has authenticated to the WebUI or CLI. The LCD panel is meant for initial setup only, and therefore is not included in the evaluated configuration. Additionally, most functions performed from the CLI can also be performed from the Web UI, with the exception of reviewing audit data. For more information on TOE management, see Section 9.1.3.

2.5.4 Protection of the TSF

The TOE is expected to ensure the security and integrity of all data that is stored locally and accessed remotely. The TOE ensures that all local system data is available to any remote trusted IT products (i.e. other TOE components). Additionally, the transmitted and received data is protected against unauthorized viewing by third parties through the use of encryption. All data transferred is monitored for changes during transmission, and integrity verification measures are taken if modifications have been detected. Time stamps are added to all audit logs and system events in order to maintain accurate records. The system clock time is kept accurate by automatically getting accurate time readings from the NTP Server to which the FireEye appliance is connected.

2.5.5 Encrypted Communications

The TOE is expected to utilize sufficient security measures to protect its data in transmission, which means it needs to utilize cryptographic methods and trusted channels. The TOE generates cryptographic keys to protect transmitted data. The TOE is also responsible for destroying these same keys when they are no longer needed.

Administrators and Monitors who access the TOE remotely rely on a trusted path to secure their communication with the TOE via the WebUI. This trusted path is established using OpenSSL 0.9.8e. OpenSSL is also used for protected communication to/from the MAX Network. Additionally, users who access the TOE via the CLI must use OpenSSH 3.8.1p1 functionality to secure their communications with the TOE. OpenSSH functionality is also used for protection of data transferred between TOE components.

2.5.6 Intrusion Detection System

The TOE monitors the network's traffic for detected malicious code, service requests, and service configuration, among other information. Anything that the TOE determines is malicious becomes an event. General information is recorded for each event, and each type of event has more specific classifications that are recorded. See Section 9.1.6 for more information on the data that is collected by the TOE.

The TOE analyzes recorded data on a statistical, signature, virtual machine, and/or heuristic basis. Each analytical result is recorded with basic information, as well as changes in the OS or network, and whether or not a buffer overflow was attempted. Administrators and Monitors are able to view the data via the WebUI or CLI. Once a threat has been detected, the system sends an alarm to the Administrator or Monitor. Depending on the deployment (inline or SPAN/tap), the TOE is also capable of dropping the traffic that was shown to represent a threat. The TOE is also capable of performing whitelisting of IPs and policy exceptions for particular keywords.

Data in the system is protected from unauthorized deletion or modification. System data is archived to a local file once the predefined number of events has been recorded to the internal database. An alarm is used to alert Administrators and Monitors of this issue.

3 Conformance Claims

3.1 CC Version

This ST is compliant with *Common Criteria for Information Technology Security Evaluation*, CCMB-2007-09-004, Version 3.1 Revision 3 July 2009.

3.2 CC Part 2 Conformance Claims

This ST and Target of Evaluation (TOE) is Part 2 conformant to the US Government Protection Profile Intrusion Detection System System for Basic Robustness Environments v.1.7. This includes all applicable NIAP and International interpretations through 07 December 2009.

3.3 CC Part 3 Conformance Claims

This ST and TOE is Part 3 conformant to the US Government Protection Profile Intrusion Detection System System for Basic Robustness Environments v.1.7, with all SARs consistent with EAL2 augmented with ALC_FLR.2. This includes all applicable NIAP and International interpretations through 07 December 2009.

3.4 PP Claims

This ST claims demonstrable conformance to US Government Protection Profile Intrusion Detection System System for Basic Robustness Environments v.1.7.

3.5 Package Claims

In addition to the US Government Protection Profile Intrusion Detection System System for Basic Robustness Environments v.1.7 being met by this TOE, SARs consistent with EAL2 augmented with ALC_FLR.2 have been claimed. Therefore, this TOE claims an assurance package for EAL2 augmented with ALC_FLR.2.

3.6 Package Name Conformant or Package Name Augmented

This ST and TOE is conformant to EAL2 package claims augmented with ALC_FLR.2.

3.7 Conformance Claim Rationale

This ST and TOE is conformant to the US Government Protection Profile Intrusion Detection System System for Basic Robustness Environments v.1.7. All SFRs stated in this ST are either conformant to CC Part 2 or the IDS System PP. All SFRs in the IDS System PP have been claimed in this ST. This ST levies the same restrictions on the TOE and the same restrictions on the operational environment of the TOE, which therefore makes the ST equivalent to the IDS System PP. According to the IDS System PP, "Intrusion Detection System System Protection Profile-conformant products support the ability that monitor (both real-time and statically) an IT System for activity that may inappropriately affect the IT System's assets and react appropriately. Intrusion Detection System System Protection Profile-conformant products also provide the ability to protect themselves and their associated data from unauthorized access or modification and ensure accountability for authorized actions. The Intrusion Detection System System Protection

Profile was constructed to provide a target and metric for the development of Systems. This ST identifies security functions and assurances that represent the lowest common set of requirements that should be addressed by a useful IDS System." Since the IDS System PP has previously been evaluated and this TOE meets a minimum standard of demonstrable conformance to the IDS PP, this TOE accurately claims conformance to the IDS System PP.

4 Security Problem Definition

4.1 Threats

The following are threats identified for the TOE and the IT System the TOE monitors. The TOE itself has threats and the TOE is also responsible for addressing threats to the environment in which it resides. The assumed level of expertise of the attacker for all the threats is unsophisticated.

4.1.1 TOE Threats

T.COMINT An unauthorized user may attempt to compromise the integrity of the data collected and produced by the TOE by bypassing a security mechanism.

T.COMDIS An unauthorized user may attempt to disclose the data collected and produced by the TOE by bypassing a security mechanism.

T.LOSSOF An unauthorized user may attempt to remove or destroy data collected and produced by the TOE.

T.NOHALT An unauthorized user may attempt to compromise the continuity of the System's collection and analysis functions by halting execution of the TOE.

T.PRIVIL An unauthorized user may gain access to the TOE and exploit system privileges to gain access to TOE security functions and data

T.IMPCON An unauthorized user may inappropriately change the configuration of the TOE causing potential intrusions to go undetected.

T.INFLUX An unauthorized user may cause malfunction of the TOE by creating an influx of data that the TOE cannot handle.

T.FACCNT Unauthorized attempts to access TOE data or security functions may go undetected.

T.EAVESDROPPING A malicious user could eavesdrop on network traffic to gain unauthorized access to TOE data.

4.1.2 IT System Threats

The following identifies threats to the IT System that may be indicative of vulnerabilities in or misuse of IT resources.

T.SCNCFG Improper security configuration settings may exist in the IT System the TOE monitors.

T.SCNMLC Users could execute malicious code on an IT System that the TOE monitors which causes modification of the IT System protected data or undermines the IT System security functions.

T.SCNVUL Vulnerabilities may exist in the IT System the TOE monitors.

T.FALACT The TOE may fail to react to identified or suspected vulnerabilities or inappropriate activity.

T.FALREC The TOE may fail to recognize vulnerabilities or inappropriate activity based on IDS data received from each data source.

T.FALASC The TOE may fail to identify vulnerabilities or inappropriate activity based on association of IDS data received from all data sources.

T.MISUSE Unauthorized accesses and activity indicative of misuse may occur on an IT System the TOE monitors.

T.INADVE Inadvertent activity and access may occur on an IT System the TOE monitors.

T.MISACT Malicious activity, such as introductions of Trojan horses and viruses, may occur on an IT System the TOE monitors.

4.2 Organizational Security Policies

An organizational security policy is a set of rules, practices, and procedures imposed by an organization to address its security needs. This section identifies the organizational security policies applicable to the IDS System PP.

P.DETECT Static configuration information that might be indicative of the potential for a future intrusion or the occurrence of a past intrusion of an IT System or events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets must be collected.

P.ANALYZ Analytical processes and information to derive conclusions about intrusions (past, present, or future) must be applied to IDS data and appropriate response actions taken.

P.MANAGE The TOE shall only be managed by authorized users.

P.ACCESS All data collected and produced by the TOE shall only be used for authorized purposes.

P.ACCACT Users of the TOE shall be accountable for their actions within the IDS.

P.INTGTY Data collected and produced by the TOE shall be protected from modification.

P. PROTCT The TOE shall be protected from unauthorized accesses and disruptions of TOE data and functions.

4.3 Assumptions

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

4.3.1 Intended Usage Assumptions

A.ACCESS The TOE has access to all the IT System data it needs to perform its functions.

A.DYNMIC The TOE will be managed in a manner that allows it to appropriately address changes in the IT System the TOE monitors.

A.ASCOPE The TOE is appropriately scalable to the IT System the TOE monitors.

A.GENPUR There are no general-purpose computing capabilities (e.g., the ability to execute arbitrary code or applications) on the TOE.

A.UPDATE The TOE will connect to the MAX Network for signature updates and to upload detected malware.

4.3.2 Personnel Assumptions

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

A.NOEVIL The authorized administrators are not careless, willfully negligent, or hostile, and will follow and abide by the instructions provided by the TOE documentation.

A.NOTRST The TOE can only be accessed by authorized users.

4.3.3 Physical Assumptions

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

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

5 Security Objectives

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

5.1 IT Security Objectives

The following are the TOE security objectives:

O.PROTCT The TOE must protect itself from unauthorized modifications and access to its functions and data.

O.IDSCAN The Scanner must collect and store static configuration information that might be indicative of the potential for a future intrusion or the occurrence of a past intrusion of an IT System.

O.IDSENS The Sensor must collect and store information about all events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets and the IDS.

O.IDANLZ The Analyzer must accept data from IDS Sensors or IDS Scanners and then apply analytical processes and information to derive conclusions about intrusions (past, present, or future).

O.RESPON The TOE must respond appropriately to analytical conclusions.

O.EADMIN The TOE must include a set of functions that allow effective management of its functions and data.

O.ACCESS The TOE must allow authorized users to access only appropriate TOE functions and data.

O.IDAUTH The TOE must be able to identify and authenticate users prior to allowing access to TOE functions and data.

O.OFLOWS The TOE must appropriately handle potential audit and System data storage overflows.

O.AUDITS The TOE must record audit records for data accesses and use of the System functions.

O.INTEGR The TOE must ensure the integrity of all audit and System data.

O.EAVESDROPPING The TOE will encrypt TSF data that traverses the network to prevent malicious users from gaining unauthorized access to TOE data.

5.2 Security Objectives for the operational environment of the TOE

The TOE's operating environment must satisfy the following objectives.

OE.AUDIT_PROTECTION	The IT Environment will provide the capability to protect audit information.
OE.AUDIT_SORT	The IT Environment will provide the capability to sort the audit information
OE.TIME	The IT Environment will provide reliable timestamps to the TOE.
OE.INSTAL	Those responsible for the TOE must ensure that the TOE is delivered, installed, managed, and operated in a manner which is consistent with IT security.
OE.PHYCAL	Those responsible for the TOE must ensure that those parts of the TOE critical to security policy are protected from any physical attack.
OE.CREDEN	Those responsible for the TOE must ensure that all access credentials are protected by the users in a manner which is consistent with IT security.
OE.PERSON	Personnel working as authorized administrators shall be carefully selected and trained for proper operation of the System.
OE.INTROP	The TOE is interoperable with the IT System it monitors.
OE.GENPUR	There are no general-purpose computing capabilities (e.g., the ability to execute arbitrary code or applications) on the TOE.
OE.UPDATE	The Operational Environment receives detected malware from the TOE and provides signature updates from the TOE on a regular basis.

6 Extended Security Functional and Assurance Requirements

6.1 Extended Security Functional Requirements for the TOE

There are no extended Security Functional Requirements for this ST that have not come from Protection Profile Intrusion Detection System System For Basic Robustness Environments Version 1.7.

6.2 Extended Security Assurance Requirements

There are no extended Security Assurance Requirements in this ST.

7 Security Functional Requirements

7.1 Security Functional Requirements for the TOE from the IDS System PP

The following table provides a summary of the Security Functional Requirements provided by the IDS System PP and are implemented by the TOE.

Security Function	Security Functional Components
Security Audit (FAU)	FAU_GEN.1 Audit data generation
	FAU_SAR.1 Audit review
	FAU_SAR.2 Restricted audit review
	FAU_SAR.3 Selectable audit review
	FAU_SEL.1 Selective audit
	FAU_STG.2 Guarantees of audit data availability
	FAU_STG.4 Prevention of Data Loss
Identification and Authentication (FIA)	FIA_UAU.1 Timing of authentication
	FIA_AFL.1 Authentication failures
	FIA_ATD.1 User attribute definition
	FIA_UID.1 Timing of identification
Security Management (FMT)	FMT_MOF.1 Management of security functions behavior
	FMT_MTD.1 Management of TSF data
	FMT_SMR.1 Security roles
Protection of the TSF (FPT)	FPT_ITA.1 Availability of exported TSF data
	FPT_ITC.1 Confidentiality of exported TSF data
	FPT_ITI.1 Integrity of exported TSF data
	FPT_STM.1 Reliable time stamps
Intrusion Detection System (IDS)	IDS_SDC.1 System Data Collection
	IDS_ANL.1 Analyzer analysis
	IDS_RCT.1 Analyzer react
	IDS_RDR.1 Restricted Data Review
	IDS_STG.1 Guarantee of System Data Availability
	IDS_STG.2 Prevention of System data loss

Table 7-1: Security Functional Requirements for the TOE from the IDS PP

7.1.1 Class FAU: Security Audit

7.1.1.1 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 auditable events for the **basic** level of audit; and

c) **Access to the System and access to the TOE and System data.**

Application Note: *The auditable events for the basic level of auditing are included in the table below.*

Component	Event	Additional Details
FAU_GEN.1	Start-up and shutdown of audit functions	None
FAU_GEN.1	Access to System	None
FAU_GEN.1	Access to the TOE and System data	Object IDS, Requested access
FAU_SAR.1	Reading of information from the audit records	None
FAU_SAR.2	Unsuccessful attempts to read information from the audit records	None
FAU_SEL.1	All modifications to the audit configuration that occur while the audit collection functions are operating	None
FIA_UAU. 1	All use of the authentication mechanism	User identity, location
FIA_UID.1	All use of the user identification mechanism	User identity, location
FMT_MOF.1	All modifications in the behavior of the functions of the TSF	None
FMT_MTD.1	All modifications to the values of TSF data	None
FMT_SMR.1	Modifications to the group of users that are part of a role	User identity
FCS_CKM.1	The object attribute(s), and object value(s) excluding any sensitive information (e.g. secret or private keys)	None
FCS_CKM.4	The object attribute(s), and object value(s) excluding any sensitive information (e.g. secret or private keys)	None
FCS_COP.1	Any applicable cryptographic mode(s) of operation, subject attributes and object attributes	None
FMT_SMF.1	All modifications in the behavior of the functions in the TSF	User identity
FTP_TRP.1	All attempted uses of the trusted path functions	User identity

Table 7-2: Auditable Events

Application Note: *Auditing cannot be enabled or disabled so startup and shutdown of the audit functions are synonymous with startup and shutdown of the TOE.*

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 identity, and the outcome (success or failure) of the event; and

7.1.1.2 **b)** For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, **the additional information specified in the** Details **column of the table above. FAU_SAR.1 Audit review**

FAU_SAR.1.1 The TSF shall provide [***Administrators***] with the capability to read [***all audit information***] 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.

Application Note: *Audit data is provided as columnar results as Linux syslog file data*

7.1.1.3 FAU_SAR.2 Restricted audit review

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

7.1.1.4 FAU_SAR.3 Selectable Audit Review

FAU_SAR.3.1 The TSF shall provide the ability to perform sorting of audit data based on date and time, subject identity, type of event, and success or failure of related event.

Application Note: *Audit data can be sorted based on regular expressions.*

7.1.1.5 FAU_SEL.1 Selective audit

FAU_SEL.1.1 The TSF shall be able to select the set of events to be audited from the set of all auditable events based on the following attributes:

a) event type;

b) [***no other attributes***]

Application Note: *"Event type" in this situation refers to the audit level of the event. For more information, refer to Table 9-1.*

7.1.1.6 FAU_STG.2 Guarantees of audit data availability

FAU_STG.2.1 The TSF shall protect the stored audit records from unauthorized deletion.

FAU_STG.2.2 The TSF shall be able to prevent modifications to the audit records.

Application Note: *The TSF disables direct access to the file system so there is no mechanism by which the audit files could be manipulated.*

FAU_STG.2.3 The TSF shall ensure that [***all audit records except those stored in the oldest log file***] will be maintained when the following conditions occur: [***audit storage exhaustion***]

Application Note: *Audit log files are restricted to a maximum compressed size of 16 MB. There are 40 such log files. When the last file is full, the oldest file is overwritten with new audit data.*

7.1.1.7 FAU_STG.4 Prevention of audit data loss

FAU_STG.4.1 The TSF shall [**overwrite the oldest stored audit records**] and send an alarm if the audit trail is full.

7.1.2 Class FIA: Identification and Authentication

7.1.2.1 FIA_UAU.1 Timing of authentication

FIA_UAU.1.1 The TSF shall allow [***no actions***] on behalf of the user to be performed before the user is authenticated.

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

7.1.2.2 FIA_AFL.1 Authentication failures

FIA_AFL.1.1 The TSF shall detect when a **settable, non-zero number** of unsuccessful authentication attempts occur related to **external IT products attempting to authenticate**.

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 product from successfully authenticating until an authorised administrator takes some action to make authentication possible for the external IT product in question.**

Application Note: *The TOE uses OpenSSL 0.9.8e to connect to the MAX Network. After the establishment of this encrypted communication, the TOE and the MAX Network mutually authenticate to each other. The threshold for the failed authentication attempts is hard coded into the TOE as one failed attempt.*

Application Note: *The TOE meets this requirement by having an Administrator (Authorized System Administrator) perform this action. "Authorized administrator" is defined as the Monitor role in the TOE, which cannot perform this action. The SFR was not changed because the text was taken directly from the PP to which the ST claims conformance. The intent of this requirement is met by alternatively assigning this authority to the Administrator.*

7.1.2.3 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) User identity;

b) Authentication data;

c) Authorisations; and

d) [***no other attributes]*.**

Application Note: *Authorizations are statically mapped to roles so the assignment of a role implicitly lists the authorizations made available to the user.*

7.1.2.4 FIA_UID.1 Timing of identification

FIA_UID.1.1 The TSF shall allow [***no actions***] on behalf of the user to be performed before the user is identified.

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

7.1.3 Class FMT: Security Management

7.1.3.1 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 of **System data collection, analysis and reaction** to authorized System administrators.

Application Note: *"Authorized System administrator" refers to the Administrator role. System data collection configuration refers to defining target IP addresses/subnet ranges to monitor and specifying proxy information if necessary. Reaction refers to what events will trigger alerts, how the alerts will be transmitted, and to whom the alerts will be sent.*

7.1.3.2 FMT_MTD.1 Management of TSF data

FMT_MTD.1.1 The TSF shall restrict the ability to query **and add System and audit data, and shall restrict the ability to query and modify all other TOE data** to [***the following roles: refer to Table 7-5***].

7.1.3.3 FMT_SMR.1 Security roles

FMT_SMR.1.1 The TSF shall maintain the **following *roles***: ***authorized administrator, authorized System Administrator, and [no other roles***].

Application Note: *"Authorized administrator" refers to the Monitor role. "Authorized System administrator", which is interpreted to be synonymous with a super user, refers to the Administrator role.*

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

7.1.4 Class FPT Protection of the TOE Security Functions

7.1.4.1 FPT_ITA.1 Availability of Exported TSF Data

FPT_ITA.1.1 The TSF shall ensure the availability of audit and System data provided to a remote trusted IT product within [***immediately upon creation of an audit record or completion of a scanning session***] given the following conditions: [***the data is stored on the appliance from which the data is being requested***].

Application Note: *Each appliance needs to be connected to the CMS manually. Once connected, the CMS polls for data from each of the appliances every 60 seconds. Connection to each appliance is done via the standard SSH based authentication mechanisms. The CMS periodically retries connection to each appliance upon failure. The data is available immediately via direct access to the appliance. However, if accessing the CMS, the data will not be available until the next CMS poll.*

Application Note: *The audit data is provided to the MAX Network for troubleshooting purposes only.*

7.1.4.2 FPT_ITC.1 Confidentiality of Exported TSF Data

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

Application Note: *The transmission of TSF data from the TSF to the MAX network is protected by HTTPS using OpenSSL 0.9.8e.*

Application Note: *The transmission of TSF data from one TOE component to another is protected by the SSH functionality in OpenSSL 0.9.8e.*

7.1.4.3 FPT_ITI.1 Integrity of Exported TSF Data

FPT_ITI.1.1 The TSF shall provide the capability to detect modification of all TSF data during transmission between the TSF and a remote trusted IT product within the following metric: [***verification of integrity using HMAC-MD5***].

FPT_ITI.1.2 The TSF shall provide the capability to verify the integrity of all TSF data transmitted between the TSF and a remote

trusted IT product and perform [***notification of update failure to Administrator***] if modifications are detected.

Application Note: *The SSL and SSH functionalities in OpenSSL 0.9.8e are used to verify the checksum of updates received from FireEye via the MAX Network and event data transferred from FireEye appliances to the CMS.*

7.1.4.4 FPT_STM.1 Reliable time stamps

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

Application Note: *System clock functionality is provided by the Linux kernel, version 2.6.22. This functionality has been unaltered from its default implementation. Using an NTP server helps timestamps be as accurate as possible.*

7.1.5 Class IDS: Intrusion Detection System Component Requirements

A family of IDS requirements was created to specifically address the data collected and analyzed by an IDS. The audit family of the CC (FAU) was used as a model for creating these requirements. The purpose of this family of requirements is to address the unique nature of IDS (system) data and provide for requirements about collecting, reviewing and managing the data. These requirements have no dependencies since the stated requirements embody all the necessary security functions.

7.1.5.1 IDS_SDC.1 System Data Collection (EXT)

IDS_SDC.1.1 The System shall be able to collect the following information from the targeted IT System resource(s):

a) [**service requests, network traffic, security configuration changes, data introduction, detected malicious code, access control configuration, service configuration, detected known vulnerabilities**]; and

b) [***no other events***]. (EXT)

Application Note: *The categories listed above refer to specific behavior as follows:*

- *Service Requests: Execution of system processes*
- *Network Traffic: HTTP traffic, potential botnet callback communications (IRC, FTP, TFTP, SMTP)*

- *Security Configuration Changes: Modifications to firewall or antivirus behavior*
- *Data Introduction: Additions to the OS file system*
- *Detected Malicious Code*
- *Access Control Configuration*
- *Service Configuration*
- *Detected Known Vulnerabilities*

IDS_SDC.1.2 At a minimum, the System shall collect and record the following information:

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

b) The additional information specified in the Details column of the table below. (EXT)

Component	Event	Details
IDS_SDC.1	Service Requests	**Specific service, source address, destination address**
IDS_SDC.1	Network Traffic	**Protocol, source address, destination address**
IDS_SDC.1	Security configuration changes	**Source address, destination address**
IDS_SDC.1	Data introduction	**Object IDS, location of object, source address, destination address**
IDS_SDC.1	Detected malicious code	**Location, identification of code**
IDS_SDC.1	Access control configuration	**Location, access settings**
IDS_SDC.1	Service configuration	**Service identification (name or port), interface, protocols**
IDS_SDC.1	Detected known vulnerabilities	**Identification of the known vulnerability**

Table 7-3: System Events

7.1.5.2 IDS_ANL.1 Analyzer analysis (EXT)

IDS_ANL.1.1 The System shall perform the following analysis function(s) on all IDS data received:
a) [**Statistical, signature**]; and
b) [***Virtual machine, heuristic***]. **(EXT)**

Application Note: *Statistical analysis looks for abnormal traffic that may be indicative of scanning or propagation. Signature analysis examines payloads for callback traffic. Virtual machine analysis replays suspected malicious traffic against a virtual machine in order to determine its effects. Heuristic analysis attempts to discover malicious data in application-based traffic (javascript, pdf, HTML).*

Application Note: *IDS data is synonymous with system data.*

IDS_ANL.1.2 **The System shall record within each analytical result at least the following information:**

a) Date and time of the result, type of result, identification of data source; and

b) [***OS behavior changes, network traffic changes, attempted buffer overflow***]. **(EXT)**

Application Note: *OS behavior changes refer to the effect that replayed traffic has on the target virtual machine, including file system and registry changes. Network traffic changes refer to any additional traffic the virtual machine attempts to transmit following the introduction of the replayed traffic. Attempted buffer overflows are detected by analyzing the instructions sent to the CPU of the virtual machine.*

7.1.5.3 IDS_RCT.1 Analyzer react (EXT)

IDS_RCT.1.1 The System shall send an alarm to [***the Administrator or Monitor via one or more of the following methods: the Web GUI, SMTP, SNMP, rsyslog, and/or HTTP POST***] and take [***the following configurable actions: log an event, send configuration updates to external IT products (i.e. firewalls), drop the traffic***] when an intrusion is detected. (EXT)

Application Note: *The appliance will only "drop the traffic" in the Malware Protection System appliance, and only if it is run in-line. This action is performed if a threat is detected on an in-line MPS appliance.*

Application Note: *The TOE may "drop the traffic" based upon policies or whitelisting of IP addresses. Policy exceptions can be defined to allow for certain information to pass through.*

7.1.5.4 IDS_RDR.1 Restricted Data Review (EXT)

IDS_RDR.1.1 The System shall provide [***Administrators and Monitors***] with the capability to read [*all data*] from the System data. (EXT)

IDS_RDR.1.2 The System shall provide the System data in a manner suitable for the user to interpret the information. (EXT)

IDS_RDR.1.3 The System shall prohibit all users read access to the System data, except those users that have been granted explicit read-access. (EXT)

7.1.5.5 IDS_STG.1 Guarantee of System Data Availability (EXT)

IDS_STG.1.1 The System shall protect the stored System data from unauthorized deletion. (EXT)

IDS_ STG.1.2 The System shall protect the stored System data from modification. (EXT)

Application Note: *Authorized deletion of data is not considered a modification of System data in this context. This requirement applies to the actual content of the System data, which should be protected from any modifications.*

IDS_STG.1.3 The System shall ensure that [***archiving of***] System data will be maintained when the following conditions occur: [**System data storage exhaustion, failure**]. (EXT)

Application Note: *After 300,000 events have been recorded to the database, the data is archived to a local file. CLI commands can be used to upload this to a separate device for archive or backup purposes.*

7.1.5.6 IDS_STG.2 Prevention of System data loss (EXT)

IDS_STG.2.1 The System shall [***overwrite the oldest stored System data***] and send an alarm if the storage capacity has been reached. (EXT)

Application Note: *FireEye systems have 500 GB of storage. When disk space is exhausted, an alarm is sent to an Administrator or Monitor. Additionally, quotas are put into place to limit how much space is allocated for the following data types: forensic data, audit logs, and archived malware. Forensic data, archived internet traffic, archived malware, and audit logs have a quota of 10 GB, 5 GB, 5 GB, and 640 MB, respectively. The oldest files are overwritten once the total storage of a data type exceeds the quota.*

7.2 Additional Security Functional Requirements for the TOE

The following table provides a summary of additional Security Functional Requirements implemented by the TOE that go above and beyond those provided by the IDS System PP. These SFRs are pulled from CC Part 2.

Security Function	Security Functional Components
Cryptographic Support (FCS)	FCS_CKM.1(1) Cryptographic key generation
	FCS_CKM.1(2) Cryptographic key generation
	FCS_CKM.4 Cryptographic key destruction
	FCS_COP.1(1) Cryptographic operation
	FCS_COP.1(2) Cryptographic operation
Identification and Authentication (FIA)	FIA_AFL.1(1) Authentication failure handling
Security Management (FMT)	FMT_SMF.1 Specification of management functions
Trusted Path/Channel (FTP)	FTP_TRP.1 Trusted path

Table 7-4: Additional Security Functional Requirements for the TOE

7.2.1 Class FCS: Cryptographic Support

7.2.1.1 FCS_CKM.1 (1) Cryptographic Key Generation

Hierarchical to: No other components.

FCS_CKM.1.1(1) The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [***AES in CBC mode***] and specified cryptographic key sizes [***128 bits***] that meet the following: [***RFC 3565***].

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or
FCS_COP.1 Cryptographic operation]
FCS_CKM.4 Cryptographic key destruction

Application note: *This SFR supports key generation for OpenSSL for use by the MAX Network and the WebUI. TLS v1.0 and SSLv3 are used by the OpenSSL libraries in FireEye. All cryptographic functions are handled by the Linux kernel.*

7.2.1.2 FCS_CKM.1 (2) Cryptographic Key Generation

Hierarchical to: No other components.

FCS_CKM.1.1(2) The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm

[***AES in CBC mode***] and specified cryptographic key sizes [***256 bits***] that meet the following: [***RFC 3565***].

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation]
FCS_CKM.4 Cryptographic key destruction

Application note: *This SFR supports key generation for the SSH functionality in OpenSSL for use by the CLI and Central Management System (CMS). All cryptographic functions are handled by the Linux kernel.*

7.2.1.3 FCS_CKM.4 Cryptographic Key Destruction

Hierarchical to: No other components.

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [***overwrite***] that meets the following: [***no standard***].

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]

7.2.1.4 FCS_COP.1 (1) Cryptographic Operation

Hierarchical to: No other components.

FCS_COP.1.1(1) The TSF shall perform [***encryption and decryption***] in accordance with a specified cryptographic algorithm [***AES in CBC mode***] and cryptographic key sizes [***128 bits***] that meet the following: [***RFC 3565***].

Dependencies: [FDP_ITC.1 Import of user data without security attributes or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

Application note: *This SFR supports encryption and decryption for OpenSSL for use by the MAX Network and the WebUI. TLS v1.0 and SSLv3 are used by the OpenSSL libraries in FireEye. All cryptographic functions are handled by the Linux kernel.*

7.2.1.5 FCS_COP.1 (2) Cryptographic Operation

Hierarchical to: No other components.

FCS_COP.1.1(2) The TSF shall perform [***encryption and decryption***] in accordance with a specified cryptographic algorithm [***AES in CBC mode***] and cryptographic key sizes [***256 bits***] that meet the following: [***RFC 3565***].

Dependencies: [FDP_ITC.1 Import of user data without security attributes or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

Application note: *This SFR supports encryption and decryption for the SSH functionality in OpenSSL for use by the CLI and Central Management System (CMS). No particular terminal or console software or hardware is supported or recommended to be used with FireEye. Any program or system with an SSH connection will work. All cryptographic functions are handled by the Linux kernel.*

7.2.2 Class FIA: Identification and Authentication

7.2.2.1 FIA_AFL.1 (1) Authentication failure handling

Hierarchical to: No other components.

FIA_AFL.1.1 (1) The TSF shall detect when an administrator configurable positive integer within [***5***] unsuccessful authentication attempts occur related to [***Administrator and Monitor authentication***].

FIA_AFL.1.2 (1) When the defined number of unsuccessful authentication attempts has been [**met**], the TSF shall [***lock out the Administrator and Monitor for 30 minutes].***

Dependencies: FIA_UAU.1 Timing of authentication

Application note: *For this SFR, "administrator" refers to the Administrator role.*

7.2.3 Class FMT: Security Management

7.2.3.1 FMT_SMF.1 Specification of Management Functions

Hierarchical to: No other components.

FMT_SMF.1.1 The TSF shall be capable of performing the following management functions: ***[see Table 7-5 Management Functions of the TOE].***

Interface	Role (s)	Allowed Functions
CLI	Administrator	Modify date and time
		Manage CLI sessions
		Configure IP address
		Create users
		Modify users
		Delete users
		Manage TOE configuration
		Log Management
		WebUI configuration
		Query system data
		Query audit data
		Upload certificates
		Download updates
		Modify the behavior of the function of system data collection
		Modify the behavior of the function of analysis
		Modify the behavior of the function of reaction
	Monitor	Query system data
		Submit potential malware for analysis
WebUI	Administrator	Query system data
		Modify the behavior of the function of system data collection
		Modify the behavior of the function of analysis
		Modify the behavior of the function of reaction
		Modify date and time
		Configure IP address
		Create users
		Modify users
		Delete users
		Manage TOE configuration
		WebUI configuration
		Upload certificates
		Download updates
	Monitor	Query system data
		Submit potential malware for analysis

Table 7-5: Management Functions of the TOE

Dependencies: No dependencies.

Application Note: *CLI is a universal backend for WebUI and LCD commands. These commands are translated into CLI commands and forwarded to that interface where they are executed.*

7.2.4 Class FTP: Trusted Paths/Channels

7.2.4.1 FTP_TRP.1 Trusted Paths

Hierarchical to: No other components.

FTP_TRP.1.1 The TSF shall provide a communication path between itself and [***remote***] users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from [**modification, disclosure**].

FTP_TRP.1.2 The TSF shall permit [**remote users**] to initiate communication via the trusted path.

FTP_TRP.1.3 The TSF shall require the use of the trusted path for [**initial user authentication, [*all remote management actions*]**].

Application Note: *The following transmissions of TSF data are protected by the following trusted paths:*

- *Remote user to TSF via Web GUI – HTTPS (OpenSSL 0.9.8e – 128-bit AES)*

7.3 Operations Defined

The requirements in this document are divided into assurance requirements and two sets of functional requirements. The first set of functional requirements, which were drawn from the Common Criteria, is designed to address the core System requirements for self-protection. The second set of requirements, which were invented and categorized by the short name, IDS, is designed to address the requirements for the System's primary function, which is IDS collection of data and responses to conclusions based upon that data.

The CC permits four functional component operations—assignment, refinement, selection, and iteration —to be performed on functional requirements. This ST will highlight the four operations in the following manner:

- Assignment: allows the specification of an identified parameter. Indicated with ***bold text and italics*** if further operations are necessary by the Security Target author;

- Refinement: allows the addition of details. Indicated with ***bold text and italics*** if further operations are necessary by the Security Target author;
- Selection: allows the specification of one or more elements from a list. Indicated with underlined text; and
- Iteration: allows a component to be used more than once with varying operations.

In addition, this ST has extended requirements, as stated in the IDS System PP v1.7. These new requirements are indicated in bold text and contain the text (EXT) in the title.

8 Security Assurance Requirements

This section identifies the Security Assurance Requirement components met by the TOE. These assurance components meet the requirements for EAL2 augmented with ALC_FLR.2.

8.1 Security Architecture

8.1.1 Security Architecture Description (ADV_ARC.1)

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

ADV_ARC.1.2D: The developer shall design and implement the TSF so that it is able to protect itself from tampering by un-trusted active entities.

ADV_ARC.1.3D: The developer shall provide a security architecture description of the TSF.

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

ADV_ARC.1.2C: The security architecture description shall describe the security domains maintained by the TSF consistently with the SFRs.

ADV_ARC.1.3C: The security architecture description shall describe how the TSF initialization process is secure.

ADV_ARC.1.4C: The security architecture description shall demonstrate that the TSF protects itself from tampering.

ADV_ARC.1.5C: The security architecture description shall demonstrate that the TSF prevents bypass of the SFR-enforcing functionality.

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

8.1.2 Security-enforcing functional specification (ADV_FSP.2)

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

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

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

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

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

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

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

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

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

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

8.1.3 Basic Design (ADV_TDS.1)

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

ADV_TDS.1.2D: The developer shall provide a mapping from the TSFI of the functional specification to the lowest level of decomposition available in the TOE design.

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

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

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

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

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

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

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

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

8.2 Guidance Documents

8.2.1 Operational user guidance (AGD_OPE.1)

AGD_OPE.1.1D The developer shall provide operational user guidance.

AGD_OPE.1.1C The operational user guidance shall describe, for each user role, the user-accessible functions and privileges that should be controlled in a secure processing environment, including appropriate warnings.

AGD_OPE.1.2C The operational user guidance shall describe, for each user role, how to use the available interfaces provided by the TOE in a secure manner.

AGD_OPE.1.3C The operational user guidance shall describe, for each user role, the available functions and interfaces, in particular all security parameters under the control of the user, indicating secure values as appropriate.

AGD_OPE.1.4C The operational user guidance shall, for each user role, clearly present each type of security-relevant event relative to the user-accessible functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.

AGD_OPE.1.5C The operational user guidance shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation.

AGD_OPE.1.6C The operational user guidance shall, for each user role, describe the security measures to be followed in order to fulfill the security objectives for the operational environment as described in the ST.

AGD_OPE.1.7C The operational user guidance shall be clear and reasonable.

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

8.2.2 Preparative Procedures (AGD_PRE.1)

AGD_PRE.1.1D The developer shall provide the TOE including its preparative procedures.

AGD_PRE.1.1C The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.

AGD_PRE.1.2C The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the operational environment in accordance with the security objectives for the operational environment as described in the ST.

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

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

8.3 Lifecycle Support

8.3.1 Use of a CM system (ALC_CMC.2)

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

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

ALC_CMC.2.3D: The developer shall use a CM system. ALC_CMC.2.1C: The TOE shall be labeled with its unique reference.

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

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

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

8.3.2 Parts of the TOE CM coverage (ALC_CMS.2)

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

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

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

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

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

8.3.3 Delivery Procedures (ALC_DEL.1)

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

ALC_DEL.1.2D The developer shall use the delivery procedures.

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

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

8.3.4 Flaw reporting procedures (ALC_FLR.2)

ALC_FLR.2.1D The developer shall document flaw remediation procedures addressed to TOE developers.

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

ALC_FLR.2.3D The developer shall provide flaw remediation guidance addressed to TOE users.

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

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

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

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

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

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

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

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

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

8.4 Security Target Evaluation

8.4.1 Conformance Claims (ASE_CCL.1)

ASE_CCL.1.1D The developer shall provide a conformance claim.

ASE_CCL.1.2D The developer shall provide a conformance claim rationale.

ASE_CCL.1.1C The conformance claim shall contain a CC conformance claim that identifies the version of the CC to which the ST and the TOE claim conformance.

ASE_CCL.1.2C The CC conformance claim shall describe the conformance of the ST to CC Part 2 as either CC Part 2 conformant or CC Part 2 extended.

ASE_CCL.1.3C The CC conformance claim shall describe the conformance of the ST to CC Part 3 as either CC Part 3 conformant or CC Part 3 extended.

ASE_CCL.1.4C The CC conformance claim shall be consistent with the extended components definition.

ASE_CCL.1.5C The conformance claim shall identify all PPs and security requirement packages to which the ST claims conformance.

ASE_CCL.1.6C The conformance claim shall describe any conformance of the ST to a package as either package-conformant or package-augmented.

ASE_CCL.1.7C The conformance claim rationale shall demonstrate that the TOE type is consistent with the TOE type in the PPs for which conformance is being claimed.

ASE_CCL.1.8C The conformance claim rationale shall demonstrate that the statement of the security problem definition is consistent with the statement of the security problem definition in the PPs for which conformance is being claimed.

8.4.2 Extended Components Definition (ASE_ECD.1)

ASE_ECD.1.1D	The developer shall provide a statement of security requirements.
ASE_ECD.1.2D	The developer shall provide an extended components definition.
ASE_ECD.1.1C	The statement of security requirements shall identify all extended security requirements.
ASE_ECD.1.2C	The extended components definition shall define an extended component for each extended security requirement.
ASE_ECD.1.3C	The extended components definition shall describe how each extended component is related to the existing CC components, families, and classes.
ASE_ECD.1.4C	The extended components definition shall use the existing CC components, families, classes, and methodology as a model for presentation.
ASE_ECD.1.5C	The extended components shall consist of measurable and objective elements such that conformance or nonconformance to these elements can be demonstrated.
ASE_ECD.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ASE_ECD.1.2E	The evaluator shall confirm that no extended component can be clearly expressed using existing components.

8.4.3 ST Introduction (ASE_INT.1)

ASE_INT.1.1D	The developer shall provide an ST introduction.
ASE_INT.1.1C	The ST introduction shall contain an ST reference, a TOE reference, a TOE overview and a TOE description.
ASE_INT.1.2C	The ST reference shall uniquely identify the ST.
ASE_INT.1.3C	The TOE reference shall identify the TOE.
ASE_INT.1.4C	The TOE overview shall summarize the usage and major security features of the TOE.
ASE_INT.1.5C	The TOE overview shall identify the TOE type.
ASE_INT.1.6C	The TOE overview shall identify any non-TOE hardware/software/firmware required by the TOE.
ASE_INT.1.7C	The TOE description shall describe the physical scope of the TOE.

ASE_INT.1.8C The TOE description shall describe the logical scope of the TOE.

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

ASE_INT.1.2E The evaluator shall confirm that the TOE reference, the TOE overview, and the TOE description are consistent with each other.

8.4.4 Security objectives (ASE_OBJ.2)

ASE_OBJ.2.1D The developer shall provide a statement of security objectives.

ASE_OBJ.2.2D The developer shall provide security objectives rationale.

ASE_OBJ.2.1C The statement of security objectives shall describe the security objectives for the TOE and the security objectives for the operational environment.

ASE_OBJ.2.2C The security objectives rationale shall trace each security objective for the TOE back to threats countered by that security objective and OSPs enforced by that security objective.

ASE_OBJ.2.3C The security objectives rationale shall trace each security objective for the operational environment back to threats countered by that security objective, OSPs enforced by that security objective, and assumptions upheld by that security objective.

ASE_OBJ.2.4C The security objectives rationale shall demonstrate that the security objectives counter all threats.

ASE_OBJ.2.5C The security objectives rationale shall demonstrate that the security objectives enforce all OSPs.

ASE_OBJ.2.6C The security objectives rationale shall demonstrate that the security objectives for the operational environment uphold all assumptions.

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

8.4.5 Derived security requirements (ASE_REQ.2)

ASE_REQ.2.1D The developer shall provide a statement of security requirements.

ASE_REQ.2.2D The developer shall provide a security requirement's rationale.

ASE_REQ.2.1C The statement of security requirements shall describe the SFRs and the SARs.

ASE_REQ.2.2C All subjects, objects, operations, security attributes, external entities and other terms that are used in the SFRs and the SARs shall be defined.

ASE_REQ.2.3C The statement of security requirements shall identify all operations on the security requirements.

ASE_REQ.2.4C All operations shall be performed correctly.

ASE_REQ.2.5C Each dependency of the security requirements shall either be satisfied, or the security requirements rationale shall justify the dependency not being satisfied.

ASE_REQ.2.6C The security requirements rationale shall trace each SFR back to the security objectives for the TOE.

ASE_REQ.2.7C The security requirements rationale shall demonstrate that the SFRs meet all security objectives for the TOE.

ASE_REQ.2.8C The security requirements rationale shall explain why the SARs were chosen.

ASE_REQ.2.9C The statement of security requirements shall be internally consistent.

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

8.4.6 Security Problem Definition (ASE_SPD.1)

ASE_SPD.1.1D The developer shall provide a security problem definition.

ASE_SPD.1.1C The security problem definition shall describe the threats.

ASE_SPD.1.2C All threats shall be described in terms of a threat agent, an asset, and an adverse action.

ASE_SPD.1.3C The security problem definition shall describe the OSPs.

ASE_SPD.1.4C The security problem definition shall describe the assumptions about the operational environment of the TOE.

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

8.4.7 TOE Summary Specification (ASE_TSS.1)

ASE_TSS.1.1D The developer shall provide a TOE summary specification.

ASE_TSS.1.1C The TOE summary specification shall describe how the TOE meets each SFR.

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

ASE_TSS.1.2E The evaluator shall confirm that the TOE summary specification is consistent with the TOE overview and the TOE description.

8.5 Tests

8.5.1 Evidence of Coverage (ATE_COV.1)

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

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

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

8.5.2 Functional Testing (ATE_FUN.1)

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

ATE_FUN.1.2D The developer shall provide test documentation

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

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

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

ATE_FUN.1.4C The actual test results shall be consistent with the expected test results.

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

8.5.3 Independent Testing - Sample (ATE_IND.2)

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

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.

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

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

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

8.6 Vulnerability Assessment

8.6.1 Vulnerability Analysis (AVA_VAN.2)

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

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

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

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

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

AVA_VAN.2.4E The evaluator shall conduct penetration testing, based on the identified potential vulnerabilities, to determine that the TOE is resistant to attacks performed by an attacker possessing Basic attack potential.

9 TOE Summary Specification

9.1 TOE Security Functions

The following sections identify the security functions of the TOE. They include Security Audit, Identification and Authentication, Security Management, Protection of the TSF, Encrypted Communications, and Intrusion Detection System.

9.1.1 Security Audit

The MPS, Malware-Analysis, and CMS instances all perform their own auditing. Each appliance audits its own behavior and stores its syslog, or audit data in its respective internal database. CMS receives detections of audit events from all other instances and can display the aggregated audit data through reports via the CLI. Audit data can only be reviewed via the CLI, and only Administrators can access the audit data. Administrators can either use the CLI on each instance of FireEye to view and sort audit data for that particular appliance, or they use the CLI on the CMS to view audit data for all FireEye appliances. Audit data can be sorted based on the following: date and time, subject identity, event type, and outcome of event. Audit data is provided to the Administrator as columnar results as Linux syslog file data.

All user actions and cryptographic actions on the TOE are audited by the CLI. The CLI is a universal backend for WebUI and LCD commands, which are translated into CLI commands and forwarded to the TOE component where they are executed, Config or Events Storage.

9.1.1.1 Audit Records

The TOE generates audit records on a given system. Reports, or audit records, are generated based on the following events:

- Start-up and shutdown of audit functions
- Access to the system
- Access to the TOE and system data
- Reading of information from the audit records
- Unsuccessful attempts to read information from the audit records
- All modifications made to the audit configuration that occur while the audit collection functions are operating
- All use of the authentication mechanism
- All use of the user identification mechanism
- All modifications in the behavior of the functions of the TSF
- All modifications to the values of TOE data
- Modifications to the group of users that are part of a roleThe object attribute(s), and object value(s) excluding any sensitive information (e.g. secret or private keys)

- Any applicable cryptographic mode(s) of operation, subject attributes and object attributes
- All modifications in the behavior of the functions in the TSF
- All attempted uses of the trusted path functions

Auditing is always functional and thus, cannot be disabled or enabled. As a result, the starting up and shutting down of audit functions is synonymous with the startup and shutdown of the TOE. Within each of the audited events listed above, the TOE records at least the date and time of the event, the type of event, the subject's claimed identity, and the outcome (success or failure) of that event. Additional attributes that the TOE records for specific events have been listed in the Additional Details Column of Table 7-2. The date and time is kept accurate from synchronization with the NTP Server.

9.1.1.2 Audit Storage

In the evaluated configuration, The TOE maintains 40 audit log files that are each restricted to a maximum compressed size of 16 MB. When syslog, or audit data reaches this maximum size, the file is backed up and a new syslog file is started. Each backup is incremented. When the last file has become full, the oldest file is then overwritten with the new audit data and an alarm notifies the Administrator that the audit trail is full. The alarm can be configured to be sent to selected users by an Administrator, and can be sent using the following mechanisms: Console, email, SNMP trap, or syslog. When syslog data is reviewed, all of the backups are included as well in the review process. The TOE protects all stored audit data from unauthorized deletion and modification. Direct access to the file system is disabled in the evaluated configuration. Therefore, there is no mechanism by which the stored audit files can be manipulated.

9.1.1.3 Audit Levels

The Administrator determines the audit level for the TOE. The audit level determines the set of auditable events which are selected to be audited. The following list displays which auditable events are logged as well as the level at which they are logged:

Event	Event Location	Level Logged At
Login/Logout	CLI via SSH	Information
Login/Logout	WebUI	Always logged and cannot be turned off
CLI Commands	CLI	Notice
WebUI Actions	WebUI	TBA
Illegal Access Attempts to commands or other parts of the system	CLI or WebUI	Warning
Authentication and Action Failures	CLI or WebUI	Error
Memory allocation errors File descriptor errors Inability to store events in the database VM creation or analysis errors Network Connection Errors	Internal	Critical

Table 9-1: Audited Events and Levels

9.1.2 Identification and Authentication

The password protected LCD panel allows the LCD administrator to perform limited configuration functions on the TOE without being identified to the TOE. Since a username is not required to authenticate to the LCD panel, it is assumed that individuals with physical access to the TOE will also be users of the TOE. Since the LCD panel is intended for initial setup of a FireEye appliance only, the LCD panel is not part of the TSF in the evaluated configuration. The FireEye operational guidance lists an alternate way to perform initial setup, as well, using the serial port and a CLI session using SSH.

Prior to performing management functions on any instance of the TOE via the WebUI or CLI, users must log into that specific instance with their username and password. Users are not allowed to perform any functions on the TOE from the WebUI or CLI without first being successfully identified and authenticated to the TOE's base Linux systems. The TOE verifies all submitted passwords by hashing them and comparing them against the hashed passwords stored in the TOE's internal database. If the password hashes match, the user is authenticated to the TOE, and his or her session is created with an assigned role. Once a user is granted the Administrator or Monitor role, he or she is then authorized to perform the TOE functions associated with that role. In the evaluated configuration, users have five attempts at authenticating to the TOE. These attempts can be made from the CLI or the Web UI. Once this threshold of unsuccessful logins has been met, irrespective of where the attempt was made from, the specified user's account will be locked out for a 30 minutes. Once this time has elapsed, that user will be allowed to login again. Both of these values can be configured by Administrators.

The TOE maintains specific information, or attributes about each user. These attributes include the following:

- User Identity - username
- Authentication Data – hashed password
- Authorizations – based on role
- User Role – Administrator or Monitor

This information is used to properly associate a user's claimed identity with their role and authorizations associated with their role.

In addition to user authentication, certificates and vendor supplied username and password credentials are used to authenticate the MAX Network to a TOE appliance and to authenticate a TOE appliance to the MAX Network. The MAX Network is denied connectivity to the TOE if the TOE cannot authenticate the identity of the MAX network. Likewise, the TOE appliance is denied connectivity if the MAX network cannot authenticate the identity of its Updates component. If the TOE and MAX network cannot authenticate to each other on the first attempt, the Administrator will address the errors on the TOE and will need to contact the vendor in order to resolve the issue.

9.1.3 Security Management

The TOE maintains two roles – Administrators and Monitors. All users are associated with roles when they receive an account on the TOE. These roles determine what functions can be performed on the TOE. Administrators have full access to all monitoring and administrative functions. Monitors are only allowed access to monitoring functions. Administrators and Monitors access the TOE via the WebUI and CLI. The following table provides a list of TOE functions that can be performed from each interface via the respective role.

Interface	Role (s)	Allowed Functions
CLI	Administrator	Modify date and time
		Manage CLI sessions
		Configure IP address
		Create users
		Modify users
		Delete users
		Manage TOE configuration
		Log Management
		WebUI configuration
		Query system data
		Query audit data
		Upload certificates
		Download updates
		Modify the behavior of the function of system data collection
		Modify the behavior of the function of analysis
		Modify the behavior of the function of reaction
	Monitor	Query system data
		Submit potential malware for analysis
WebUI	Administrator	Query system data
		Modify the behavior of the function of system data collection
		Modify the behavior of the function of analysis
		Modify the behavior of the function of reaction
		Modify date and time
		Configure IP address
		Create users
		Modify users
		Delete users
		Manage TOE configuration
		WebUI configuration
		Upload certificates
		Download updates
	Monitor	Query system data
		Submit potential malware for analysis

Table 9-2: TOE Functions by Role and Interface

Note: For a more comprehensive list of what Administrators and Monitors can do on the TOE, see Table 7-5 Management Functions of the TOE.

9.1.3.1 Command Line Interface

All communication via the CLI and the TOE are secured with the SSH functionality in OpenSSL 0.9.8e. A terminal application which supports SSH such as PuTTy is used to access the CLI remotely. The CLI can be accessed locally via a serial port and a similar terminal application. Only Administrators can access the TOE administrative functions via the CLI. Monitors are not allowed access to the administrative functions. Administrators use any application able to initiate an SSH connection to connect to the CLI. The CLI can operate under different modes, as stated in the table below. An Administrator can enter any mode in order perform management functions on the TOE. Table 9-3 shows each mode and how they are used.

Mode	Description
Standard	Monitor system operation and issue some system commands, such as ping and traceroute. This is the default login mode.
Enabled	Set up and monitor the system (includes all commands in the Standard mode). To access the Enabled mode, enter "enable" in the Standard mode.
Configuration	Configure the FireEye application (includes all commands). To access Configuration mode, enter "config terminal" in the Enabled mode.

Table 9-3: Command Line Interface Modes

9.1.3.2 Web User Interface (WebUI)

Once logged into the WebUI with username and password, Administrators and Monitors are presented with the Dashboard landing page. The Dashboard provides a simple overview of appliance and network status. As specified below, there are some options that are only available to Administrators using the Dashboard. The following options are available through the Dashboard:

- **Alerts** – Provides Administrators and Monitors with the ability to set an alert policy for the two basic event types - Infection and Malware Callback events.
- **Summaries** - provides general information about infections. The TOE allows Administrators and Monitors to view infection in one of three forms – chart, tree map, or malware view. In the Charts view, which is the default view, bar charts are presented to illustrate the top infected hosts and top malware events. In the Tree Maps view, box sizes and colors are used to represent the number of infections and their severity. The Malware view displays summary information in table form.

- **Filters** - Administrators and Monitors can define global filters that apply to all of the monitoring pages. These filters affect only the display of data, not the data itself. Filters can be saved for future use and can be defined according to the following criteria:
    - Source/Destination IP Address
    - Time Period
    - Event Type

- **Appliance Health** – This tab provides an overview of the connectivity status, version, and time settings of each configured appliance. This tab is only available on a CMS FireEye appliance.

- **Malware Analyses** – This tab performs analyses on files or locations that have already been identified as infected. This tab is visible only for FireEye CMS and the FireEye Malware Analysis appliance configuration.

- **Appliance Settings** – Only Administrators, through the Appliance Settings tab, can modify settings for communication and other appliance settings. Items that can be modified through this tab are:

    - Date/Time - The date and time can be set manually or one or more NTP servers can be specified to synchronize the time.

    - User Accounts - User accounts can be created with roles being assigned for appliance access.

    - Email - Email settings can be configured for administrative events, e.g. automatic update of security contents, security warnings, etc.

    - MAX Network - Settings can be configured for communication and notifications over the FireEye Malware Exchange Network (MAX).

    - Notifications - Allows event alert policy and malware analysis alerts to be set. These notifications can be sent by any of the following methods:

        - Email
        - HTTP
        - rsyslog
        - SNMP

    - Network – Allows network settings such as IP address, subnet mask, and default gateway DNS server address to be set.

    - Network Ranges – Allows a range of designated IP addresses be captured or discarded.

    - Malware Analysis Configuration - Parameters can be defined for malware analysis for files or locations that have already been identified as having infections.

- Guest Images - The guest images page lists the guest virtual victim machine images that the appliance loads to evaluate the effects of suspicious traffic on the devices in a network.
- SSL Certificates - SSL certificate information for certificate authentication can be uploaded in order to access the TOE. By default the TOE provides self-signed certificates. However, this has the potential to cause errors with some browsers. The ability to upload a certificate is provided to mitigate the risk of this error occurring.
- Appliance Update - The latest software updates/upgrades can be uploaded (requires a reboot).

- **CMS Settings** – This tab allows Administrators to configure settings for the FireEye CMS. This tab is only available on the CMS FireEye appliance. The functionality is similar to the Appliance Settings tab.
- **Reports** - The Reports tab allows Administrators to manually generate reports based on attacks, attackers, and botnet activity. Reports can be scheduled for automatic generation and email distribution. Monitors cannot use the Reports tab, even though it primarily involves the viewing of System data. However, this data is available through other tabs in the FireEye WebUI, so the involved security functional requirement is not violated.

Figure 6 provides a snapshot of the Dashboard view when a user initially logs into the TOE via the WebUI.

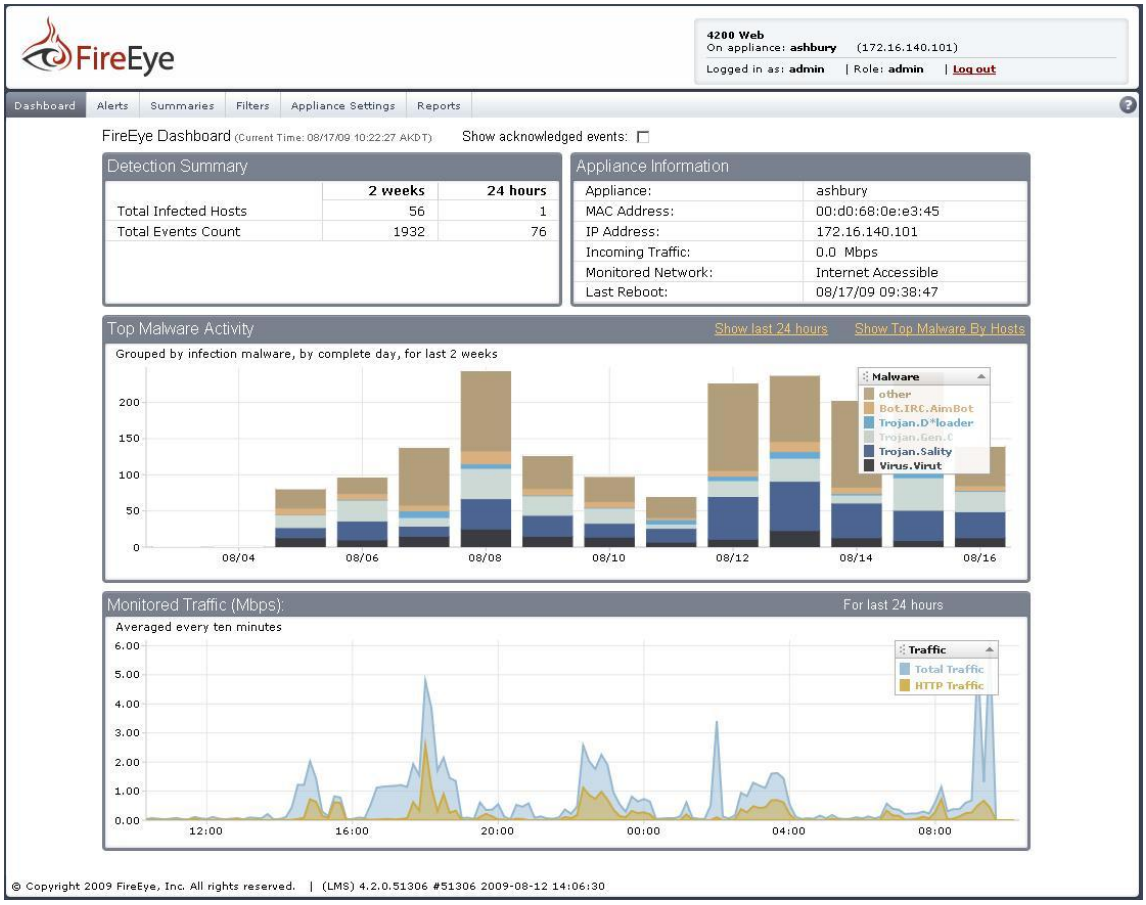

Figure 6 – Screenshot of FireEye Dashboard

9.1.4 Protection of the TSF

The TOE authenticates to the MAX Network with certificates and with a username and password provided by the vendor that is configured by the administrator during initial setup. All communication to/from the MAX network is protected from modification and disclosure and is secured using OpenSSL 0.9.8e using 128 bit AES keys. The TOE and the MAX Network have the ability to verify the integrity of all transferred data to/from the MAX Network by creating a hash using HMAC-MD5. If an MD5 hash check fails, or if modifications to the file are detected, a notification of update failure is sent to the Administrator via one or more of the following methods: the Web GUI, SMTP, SNMP, rsyslog, and/or HTTP POST. All communication to/from the TOE components is protected from disclosure by and the SSL and SSH functionalities in OpenSSL 0.9.8e and OpenSSH 3.8.1p1 using 128-bit and 256-bit AES keys. For more information regarding the situations in which types of encryption are used refer to section 9.1.5. To protect TSF data from modification, the integrity of all data transferred between TOE components is verified using HMAC-MD5 secure hashing. If an MD5 hash check fails, or if modifications to the file are detected, a notification of update failure is sent to the Administrator via one or more of the following methods: the WebUI, SMTP, SNMP, rsyslog, and/or HTTP POST.

All audit and system data is available to the other TOE components once the scanning session has been completed, as long as the audit data is stored on the local system. The TOE pushes the data to the MAX Network for troubleshooting purposes only. This push action can be enabled or disabled by Administrators using the CLI and is enabled by default in the evaluated configuration. This action can be done during runtime or during install. Since the TOE consists of distributed pieces of hardware, the availability of data that is transmitted between appliances is an important consideration. The CMS is configured to receive event data from the MPS and Malware Analysis products for the purpose of aggregation and single-point reporting. Once this configuration is established, the CMS polls the other appliances once a minute. If a connection cannot be made, the CMS will continue polling. TSF data which is to ultimately reside in the CMS is made available immediately upon its receipt from the other appliances.

9.1.5 Encrypted Communications

Transmitting data to/from remote users and systems requires encryption so data cannot be accessed or modified by anyone during transmission. The TOE generates cryptographic keys for encryption and decryption of communication traffic to/from the CLI, WebUI, MAX Network, and for communication between the CMS and the remainder of the TOE.

For communications from user space (i.e. a web browser or terminal software) to the TOE for the CLI and WebUI, cryptography is used to establish a trusted path that protects user activity from disclosure. These trusted paths are initiated by the Administrator or Monitor before they send their authentication request to the TOE and are used during their management of the TOE.

The Linux OS and OpenSSL 0.9.8e and OpenSSH 3.8.1p1 libraries and modules generate the keys for cryptographic operations. The OpenSSL library uses TLS v1.0 and SSLv3. The following interfaces use the following encryption standards and transfer protocols:

- Administrator to TSF using the Command Line Interface (CLI) and TOE to TOE – SSH – OpenSSH 3.8.1p1 256-bit AES in CBC mode, conformant to RFC 3565 – used for establishment of the trusted path and protecting communications between the CMS and other appliances

- User to TSF via Web GUI and TSF to MAX Network – HTTPS – OpenSSL 0.9.8e 128-bit AES in CBC mode, conformant to RFC 3565 – used for establishment of the trusted path and downloading updated analysis files

The external interface connecting the network to the TOE using a Port Mirror does not transfer any data externally, so it requires no encryption. The TOE destroys keys that have already been used by overwriting them with the newly generated keys. The TOE uses these keys to encrypt data it transfers and allows the keys required to decrypt the data only to authenticated users. Certificates are used as supported by OpenSSL.

9.1.6 Intrusion Detection System

The TOE collects the following information from the Monitored Network: service requests, network traffic, security configuration changes, data introduction, detected malicious code, access control configuration, service configuration, and detected known vulnerabilities. Table 9-4 specifies additional information collected for each event.

Event	Information Recorded
Service requests	Execution of system processes, Specific service, source address, destination address
Network traffic	HTTP traffic, potential botnet callback communications (IRC, FTP, TFTP, SMTP), Protocol, source address, destination address
Security configuration changes	Modifications to firewall or antivirus behavior, Source address, destination address
Data introduction	Additions to the OS file system, Object IDS, location of object, source address, destination address
Detected malicious code	Location and identification of code
Access control configuration	Location and access settings
Service configuration changes	Service identification (name or port), interface, and protocols
Detected known vulnerabilities	Identification of the known vulnerability

Table 9-4: System Event Information Recorded

Each licensed FireEye appliance gathers all of the above information. This information is passed to CMS when the appliances are polled. For each of these information types, general information is collected and recorded:

- Date and time of event
- Type of event
- Subject ID
- Outcome (success or failure)

The TOE is also capable of defining policies while running in in-line mode to drop packets or performing whitelisting. Policies can be configured to allow all traffic to bypass through a filter and pass through the appliance or monitor all traffic and drop traffic that is found to be malicious. Whitelisting can also be configured to only allow certain IP addresses to be accessed by the network monitored by the TOE. This is performed to let only those areas which are trusted to communicate with one another.

Policy Exceptions can also be configured to allow information with certain keywords through for analysis.

9.1.6.1 Analysis

Once an hour, an internal TOE component polls its database to determine if any new anomalies have been discovered. If an anomaly is discovered, the Analysis Environment component establishes secure communication with the MAX Network over HTTPS in order to report the anomaly as a zero-day attack to FireEye. The data that is sent to the MAX Network about discovered threats includes the following: name of threat discovered (if previously known), count of said threat attacks/occurrences, specific malicious URLs, specific botnet or command and control server IPs, signature of threat detected.

Once a week, the TOE polls the MAX Network to determine if any new updates are available. New updates are downloaded and are propagated to the TOE's internal files. These updates are generated by FireEye based on new vulnerability research as well as an examination of zero-days they receive from instances of the TOE. The integrity of all received data is verified by the TOE through the use of HMAC-MD5 secure hashing. The MPS and Malware-Analysis instances of FireEye analyze the information and determine if an alarm needs to be sent to the Administrator. These processes are described in the sections below.

For every sequence of suspicious traffic, the TOE may perform the following functions:

- Statistical analysis – looks for abnormal traffic that may be indicative of scanning or propagation.
- Content analysis – examines payloads for known callback traffic. The purpose of the MAX Network is to help keep the definitions of known callbacks up-to-date.
- Virtual machine (VM) analysis – replays suspected malicious traffic against a virtual machine in order to determine its effects.
- Heuristic analysis – attempts to discover malicious data in application-based traffic (e.g. JavaScript, PDF, and HTML).

Each analytical result captures the date and time of the result, result type, identification of the data source, OS behavior changes, network traffic changes, and attempted buffer overflow. The MPS and Malware-Analysis appliances both capture this information.

When VM analysis is performed, the specific effects which are examined are network callbacks and OS behavior changes (or infections). Callbacks refer to the system receiving traffic that causes it to respond on the network, such as a botnet controller telling a zombie host to DoS a target. OS behavior changes refer to the effect that replayed traffic has on the target virtual machine itself, including file system and registry changes. Network traffic changes refer to any additional traffic the virtual machine attempts to transmit following the introduction of the replayed traffic. Attempted buffer overflows are detected by analyzing the instructions sent to the CPU of the virtual machine.

When analysis results are uploaded to the MAX network to enhance future analysis, no user or environment information is contained within the data. The user and environment information that provided the data will remain undisclosed to others.

9.1.6.1.1 Web Infection Analysis

This analysis process is performed on the MPS appliance. When network traffic enters the TOE, it is captured through the kernel, where it is examined for statistical and heuristic anomalies which may be indicative of browser exploits. Heuristics include conventions such as hidden text, header data, deprecated or rarely-used functions, and suspicious UI elements such as 0x0 bitmap images. The following types of traffic are examined: HTML, JavaScript, images, Flash, and PDF. The Signature Matching component performs statistical analysis by running analysis tests based on historical patterns and determining whether or not the incoming traffic is suspicious. It also performs signature analysis to determine if the traffic is indicative of a known botnet.

If suspicious traffic is detected, it is sent to the Analysis Environment, which utilizes a Virtual Machine (VM) and Virtual Network (VN). The Analysis Environment simulates a web server, connects to the available VM, and has the VM simulate user interaction with the suspicious object in order to determine its effect. The Analysis Environment observes the results of this process and looks for OS changes, callbacks, and buffer overflows. If a callback that results from accessing an object is detected, it is once again examined against existing signatures to see if the object is a vector for a known botnet. The reason signature analysis is performed twice is because the initial traffic may not be representative of communications with a known botnet, but how a compromised machine responds to the traffic may be.

Event information is stored internally by the TOE and is time stamped. Event information includes the type of result and the identification of the data source. The Alerts component checks the Config component to see what type of alarm method to use to send the alarm. Once this is determined, the Alarm component sends an alarm to an Administrator and/or Monitor via one or more of the following methods: the WebUI, a notification via email, to a remote machine via SNMP trap, to rsyslog, or to HTTP Post about the threat. In addition to sending an alert, the FireEye appliance can also send out configuration updates to external IT products, specifically firewalls, that can quarantine or lockout an infected machine on the network or block traffic from a specific malicious IP. This functionality assists in the ability to take preventative measures against attacks. If the traffic is not suspicious, it is discarded by the TOE. If the TOE is deployed inline with network traffic, a positive infection result can also cause the TOE to drop the traffic which caused the infection.

9.1.6.1.2 OS Infection Analysis

This analysis process is also performed on the MPS appliance. When network traffic enters the TOE, it is captured through the kernel, where it is examined for statistical and heuristic anomalies which may be indicative of worm infection. These anomalies can

include large volumes of traffic being sent or a host replaying traffic which it has just received (an example of a worm infecting a host and attempting to propagate itself to another host in the same manner). The TOE uses signature matching library to perform basic signature analysis of packet data. If the data is flagged as suspicious, an internal TOE component is notified for prioritization and attack categorization. For example, suspected Blaster worm traffic would be flagged here so that if the VM replay triggers anomalous behavior which can be categorized as the result of a Blaster worm infection, the suspicion can be quickly confirmed.

The Analysis Environment connects to the available VM and replays the traffic to the VM. The Analysis Environment observes the results of the traffic and looks for OS changes, callbacks, and buffer overflows. The matched signature information is examined in order to categorize the attack if it is a known worm. Event information is stored internally by the TOE and is time stamped. Event information includes the type of result and the identification of the data source. The Alerts component checks the Config component to see what type of alarm method to use to send the alarm. Once this is determined, the Alarm component sends an alarm to an Administrator and/or Monitor via one or more of the following methods: the WebUI, a notification via email, to a remote machine via SNMP trap, to rsyslog, or to HTTP POST about the threat. Similar to the Web Infection Analysis detailed previously, the FireEye appliance can send out configuration updates to external devices to prevent further damage from an infection, and if the appliance is deployed in-line, the incoming traffic will be dropped.

9.1.6.1.3 Malware Infection Analysis

The way data enters the system for this analysis process is slightly different than for the MPS. Instead of capturing data constantly over a SPAN/mirror port, an Administrator or Monitor can specify the location of suspected malware to be analyzed by the TOE. This is done either by providing a link to the location or by an Administrator uploading a file via USB. Priority, target operating systems, timeout value, target browsers, and force re-analysis can also be specified. The TOE maintains checksums of submitted malware both from its own database of previously reviewed data and updates previously received from the MAX network, and will not analyze it again unless explicitly stated by an Administrator or Monitor. Suspected malware is stored in a quarantine zone in the file system. If the suspected malware is a binary that was references as a link, it is downloaded and stored in the file system. Information about the malware is also stored in the TOE's internal database, and is then queried to determine if this information was previously categorized as malicious or benign. This is known as heuristic analysis. If there is a match based on checksum, no further processing is done by the TOE.

If there is no match, the Analysis Environment connects to an available VM and instructs the VM to execute the object. If the object is a binary, it is executed in the VM's OS. If the object is a URL, the Analysis Environment provides the VM with a live network connection and instructs the VM to open a web browser and navigate to the URL. Note that a virtual network is not used in this case because the data to be analyzed has not already been received by the TOE, unlike in web infection analysis.

The Analysis Environment observes the results of the traffic and looks for OS changes, callbacks, and buffer overflows. If a callback that results from running the suspected malware is noticed, it is examined against existing signatures to see if the object is a vector for a known botnet. Event information is stored internally by the TOE and is time stamped. Event information includes the type of result and the identification of the data source. The Alerts component checks the Config component to see what type of alarm method to use to send the alarm. Once this is determined, the Alarm component sends an alarm to an Administrator and/or Monitor via one or more of the following methods: the WebUI, a notification via email, to a remote machine via SNMP trap, to rsyslog, or to HTTP Post about the threat. Multiple alarm methods can be configured for the same type of event, such as infections or callbacks, and each type of intrusion can have a different alarm method, such as email, SNMP, HTTP, or syslog. Each user that is sent alerts can configure their own settings for alert receipt.

9.1.6.2 Review of System Data

All users of the TOE can view the IDS data in the FireEye system. The Alerts tab in the WebUI or the related CLI command will display all of the Infections and Callbacks that FireEye has detected within a specified timeframe. Callback events are generated when the appliance observes outbound communications associated with a remote commands and control server, indicating that there is an established connection between an infected host and its C&C server. This could include botnet command and control communications, uploads of confidential information as well as downloads of secondary payloads (such as keyloggers or spyware). Infection events indicate that the appliance has detected a host during the process of infection by some malware. The appliance has either observed the full infection, including the exploitation of some application vulnerability along with a malware/payload transfer to the host, or it has simply observed a malware binary being transferred to the host via an unknown exploit. The exploits could be a web-browser based exploit or a network-services based exploit. For zero-day attacks the appliance displays a generic name specified to the type of attack that was performed.

The data sets that are displayed for infected hosts are: infected host IP, number of infections and callbacks, last seen at, host name, last ack, and last ack at. For botnets, the following information is displayed: botnet server, location, number of events, number of bot hosts, and last seen at.

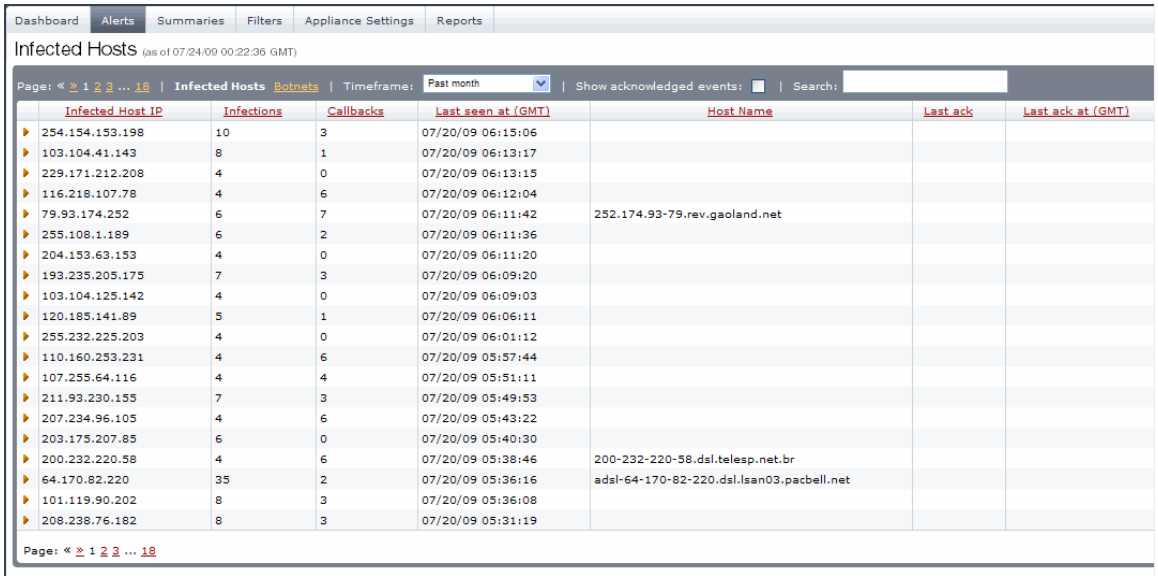

Figure 7 – Infected Hosts View

Selecting a botnet from this list displays the service ports, IP protocols, first seen timestamp, the number of VM-verified hosts, as well as the IP for each, and the number of callback hosts. Selecting an infected host shows the following information for each malware detected on that host: total events, VM-verified infections, callbacks, botnets, CnC (command and control) server, location, first seen, last seen, ports used, and protocols used, as well as initial infection URLs and assorted stats about said URLs.

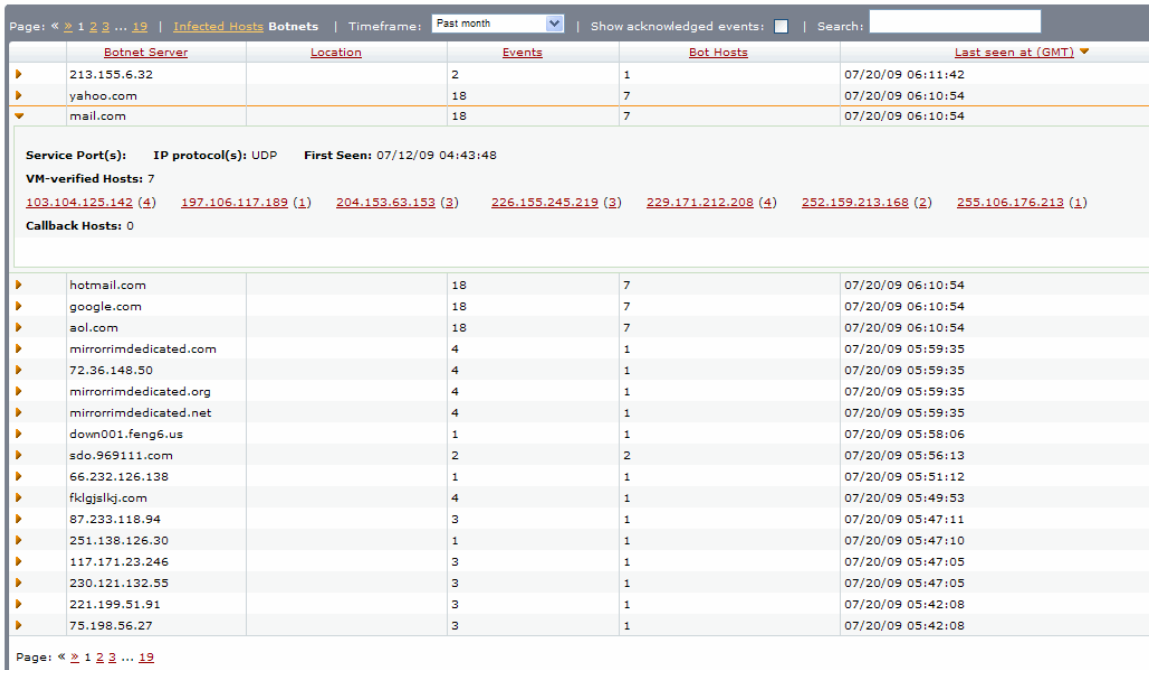

Figure 8 – Botnets View with Selected Botnet

The Summaries tab in the WebUI and related CLI command allows users to see a comprehensive list of all detected malware in a specified timeframe. Each unique piece of malware details the number of events, sources, and targets, and a timestamp of the first and last event that said malware was seen. Additionally, propagation forensics can be performed on each propagating host, displaying the following information for each propagating host: first attacked at, first attacked by, propagation delay, and first outbound attack at. Selecting any of these hosts brings up a list of attacked IPs, the direction, number of events, first and last attack timestamps, and the time span between them.

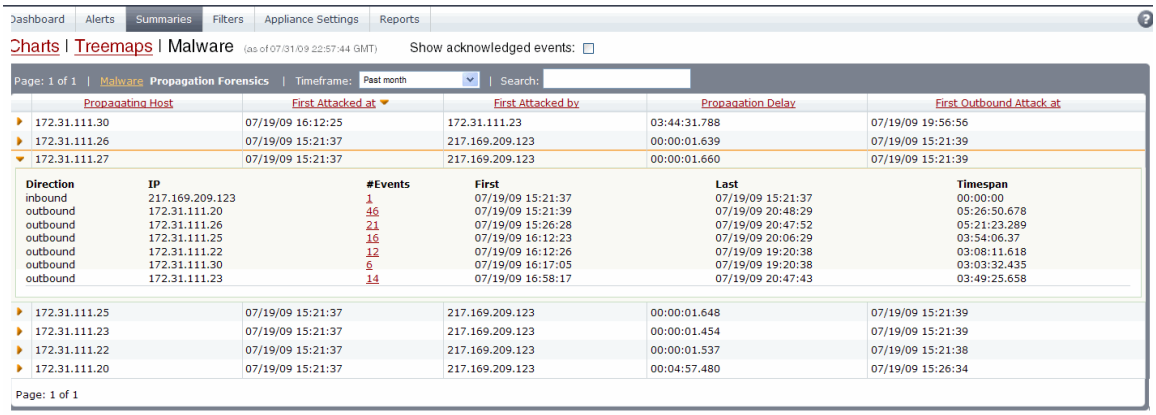

Figure 9 – Propagation Forensics view

In addition to these summaries, there are charts for top infected hosts and top malware events in this Summaries tab. There are tree maps for malware by host and malware by signature as well.

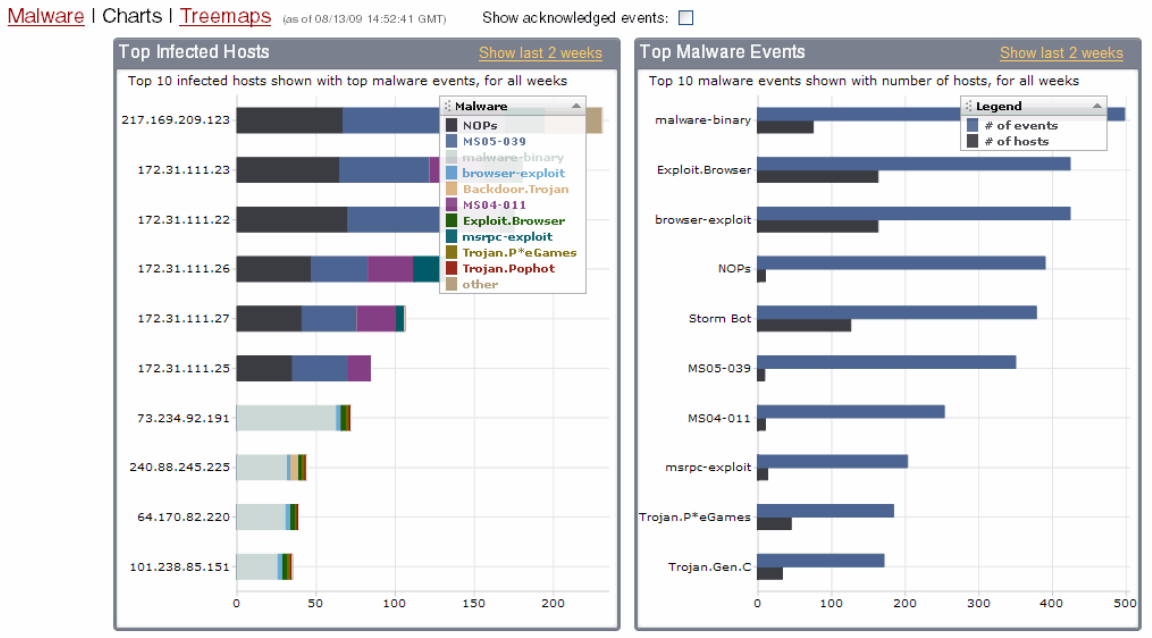

Figure 10 – Top Infected Hosts and Malware Events Charts

Administrators have multiple report templates available for use when viewing attack, attacker, and botnet activity information. Monitors cannot access the reports, but do have access to all System data through other methods. The TOE allows Administrators and Monitors to select which of the two available templates to view the report data in – an Infection Activity Report or Malware Callback Activity Report. Reports are produced in either CSV (Microsoft Excel) or PDF (Adobe Acrobat Reader) format. Reports can be automatically emailed according to a recurring schedule by Administrators. In order to do so, however, Administrators must access the report generation command line interface (CLI). Users can review all System records using either the CLI or WebUI. When viewing these records from the WebUI, Administrators and Monitors can filter the records based on criteria such as IP range, event type, and data range. When sorting the records, Administrators can sort the results based on regular expressions.

The TOE does not allow for stored system data to be deleted or modified by any users of the TOE. In the event of system data storage exhaustion or failure, the TOE ensures that said data is archived. Specifically, after 300,000 events have been logged in the database, the data is archived to a local file. Command-line Interface commands can be used to upload this to a separate device for archive or backup purposes. In the event of storage capacity being reached, an alarm is sent to an Administrator and/or Monitor via one or more of the following methods: the WebUI, a notification via email, to a remote machine via SNMP trap, to rsyslog, or to HTTP Post. FireEye appliances have about 500 GB of storage, so storage is rarely seen as an issue. Additionally, there are quotas on the amount of data each of the following data types can store: forensic data, audit logs, archived traffic, archived malware. Forensic data, archived traffic, archived malware, and audit logs have a quota of 10 GB, 5 GB, 5 GB, and 640 MB, respectively. These amounts can be configured by an Administrator. Once the total storage of data in those types exceeds the quota, the oldest logs are overwritten.

9.2 TOE Summary Specification Rationale

This section identifies the security functions provided by the TOE mapped to the security functional requirement components contained in this ST. This mapping is provided in the following table.

Security Function	Security Functional Components
Security Audit (FAU)	FAU_GEN.1 Audit data generation
	FAU_SAR.1 Audit review
	FAU_SAR.2 Restricted audit review
	FAU_SAR.3 Selectable audit review
	FAU_SEL.1 Selective audit
	FAU_STG.2 Guarantees of audit data availability
	FAU_STG.4 Prevention of Data Loss
Cryptographic Support (FCS)	FCS_CKM.1(1) Cryptographic key generation
	FCS_CKM.1(2) Cryptographic key generation
	FCS_CKM.4 Cryptographic key destruction
	FCS_COP.1(1) Cryptographic operation
	FCS_COP.1(2) Cryptographic operation
Identification and Authentication (FIA)	FIA_UAU.1 Timing of authentication
	FIA_AFL.1 Authentication failure handling
	FIA_AFL.1(1) Authentication failure handling
	FIA_ATD.1 User attribute definition
	FIA_UID.1 Timing of identification
Security Management (FMT)	FMT_MOF.1 Management of security functions behavior
	FMT_MTD.1 Management of TSF data
	FMT_SMF.1 Management Functions
	FMT_SMR.1 Security roles
Protection of the TSF (FPT)	FPT_ITA.1 Availability of exported TSF data
	FPT_ITC.1 Confidentiality of exported TSF data
	FPT_ITI.1 Integrity of exported TSF data
	FPT_STM.1 Reliable time stamps
Trusted Paths/Channels (FTP)	FTP_TRP.1 Trusted path
Intrusion Detection System (IDS)	IDS_SDC.1 System Data Collection
	IDS_ANL.1 Analyzer analysis
	IDS_RCT.1 Analyzer react
	IDS_RDR.1 Restricted Data Review
	IDS_STG.1 Guarantee of System Data Availability
	IDS_STG.2 Prevention of System data loss

Table 9-5: Security Functional Components for the TOE

9.2.1 Security Audit

The audit functions of the TOE enforce the FAU_GEN.1, FAU_SAR.1, FAU_SAR.2, FAU_SEL.1, FAU_STG.2, and FAU_STG.4 requirements.

Section 9.1.1 details how the TOE collects information about auditable events. The examination of the TSS showed that each of these requirements was successfully mapped to the SFRs listed in the Introduction section of the ST.

The generation of audit data (FAU_GEN.1.1) is provided in Section 2.5.1 of the Introduction, as well as in the section 9.1.1 of the TSS. In addition to the generation of audit data, Section 9.1.1 discusses the specific events that are audited as well as what information is recorded from each record. FAU_GEN.1.2 is further fulfilled in Section 9.1.1 with the mapping of information audited in relation to the event that is occurring. Section 2.5.1 covers the same information, albeit at a high-level.

FAU_SAR.1, FAU_SAR.2, and FAU_SAR.3 are covered in the TSS (Section 9.1.1). This section discusses that only Administrators can view audit data and this can only be done via the CLI. Monitors do not have access to the CLI administrative functions and therefore cannot view audit data. FAU_SAR.3 is also discussed in Section 9.1.1 where it states that only Administrators have the ability to sort the audit data. This section demonstrates the use of roles to apply restrictions on auditing.

FAU_STG.2 is covered in Section 9.1.1 with the discussion of ensuring that the audit data is not able to be modified or deleted by any users. The TSF, as stated in Section 9.1.1, also ensures that the most recent audit logs are maintained when audit storage has been exceeded. Finally, FAU_STG.4.1 is covered in Section 9.1.1 with the discussion of alarms being sent in the event that the audit trail has reached capacity as well as overwriting the oldest stored audit records.

9.2.2 Identification and Authentication

The Identification and Authentication function of the TOE enforces the FIA_AFL.1, FIA_AFL.1 (1), FIA_UAU.1, FIA_ATD.1, and FIA_UID.1 requirements.

FireEye uses the concept of roles and authentication data to determine a user's system access rights and operations capable of being performed.

In the Introduction, Section 2.5.2 discusses the basic overview of the identification and authentication requirements and is covered in further detail in the subsequent sections of the TSS, which are discussed below.

Section 9.1.2 discusses the primary attributes for users of the TOE. These attributes serve to identify and assign the proper abilities to users. Information such as user identity (username), authentication data (password), authorizations (based on role), and user role (Administrator or Monitor) are used in this determination of how *much* access a user has on the system. This information supports the FIA_ATD.1 requirement.

Contained in the same section, is a discussion on users not being able to perform any actions on the TOE unless they are both identified and authenticated. The LCD interface allows users to perform initial setup actions and is not part of the TSF as a result. This information supports the FIA_UAU.1 and FIA_UID.1 requirements.

Finally, in Section 9.1.2, the TOE employs the ability to lock out users once their number of unsuccessful login attempts has been exceeded. As a result, the FIA_AFL.1 requirements have been met.

9.2.3 Encrypted Communications

The security management function of the TOE enforces the FCS_CKM.1 (1), FCS_CKM.1 (2), FCS_CKM.4, FCS_COP.1 (1), FCS_COP.1(2), and FTP_TRP.1 requirements.

Section 2.5.5 discusses the TOE's need for secured communications for TOE users and for the MAX Network. Section 9.1.5 states that OpenSSH 3.8.1p1 with 256 bit AES keys is used for communication to/from the CLI and between TOE components. Additionally, OpenSSL 0.9.8e with 128 bit AES keys is used for communication to/from the WebUI and to/from the MAX network.

Remote TOE users rely on these trusted paths and channels to ensure that all communication to/from the TOE is not modified or disclosed to unauthorized users.

9.2.4 Security Management

The security management function of the TOE enforces the FMT_MOF.1, FMT_MTD.1, FMT_SMF.1 (1), FMT_SMF.1 (2), and FMT_SMR.1 requirements.

Security Management is required in order to manage the users, the roles associated with these users, and other data associated with the TOE. Security Management, consequently, supports identification and authentication, security audit, and intrusion detection system requirements.

Section 2.5.3 of the Introduction provides a general overview of the Security Management requirements as shown above and is further expounded upon in the TSS. The Introduction, TSS, and SFRs can all be mapped and interpreted through these sections.

The security management requirements as outlined by the TOE are covered by the TSS in Section 9.1.3. The main paragraph of this section identifies the division of roles into Administrator and Monitor and what functions each role is authorized to perform from each interface. The discussion of roles clearly supports the FMT_SMR.1 requirement. Sections 9.1.3.1, 9.1.3.2, and 9.1.3.3 clearly delineate these abilities that are capable of being performed under the Administrator and Monitor roles. This supports the FMT_MTD.1requirement.

9.2.5 Protection of the TSF

The protection of the TSF involves the FPT_ITA.1, FPT_ITC.1, FPT_ITI.1, and FPT_STM.1 requirements. Section 2.5.4 discusses how all TSF data transmitted from the TSF is secured and cannot be modified.

Section 9.1.4 explains that the TSF protects all externally transmitted data from unauthorized disclosure. This comes in the form of secure transfer and encryption protocols. The TSF detects modification of all TSF data during transmission between the TSF and the MAX network, and is able to verify a remote trusted IT product, and that the TSF shall provide the capability to verify the integrity of all TSF data transmitted between the TSF and a remote trusted IT product.

9.2.6 Intrusion Detection System

The IDS-specific requirements are as follows: IDS_SDC.1, IDS_ANL.1, IDS_RCT.1, IDS_RDR.1, IDS_STG.1, and IDS_STG.2. Section 2.5.6 of the ST addresses each of these requirements at a high-level and the TOE Summary Specification Section 9.1.6 shows much more detail about how each requirement is satisfied. The IDS requirements show the most basic functionality that every IDS must include and are part of the U.S. Government Protection Profile Intrusion Detection System System For Basic Robustness Environments v.1.7.

IDS_SDC.1.1 states that the System shall be able to collect information from the targeted IT System resources. Section 9.1.6 of the TSS shows that the system collects information from the monitored system and the TSS shows exactly what information is recorded. IDS_SDC.1.2 states that at a minimum, the System shall collect and record general information for all events and specific information for different types of events. Section 9.1.6 of the TSS shows that each event has general information recorded as well as specific information based on what type of event it is.

IDS_ANL.1.1 states that the System shall perform analysis function(s) on all IDS (system) data received. Section 9.1.6.1 of the TSS shows that the data is analyzed on a statistical, signature, VM, and heuristic basis. IDS_ANL.1.2 states that The System shall record within general information within each analytical result. Section 9.1.6.1 of the TSS shows that each analysis result will also be recorded with additional identifying and clarifying information, which are defined in that section.

IDS_RCT.1.1 states that the System shall send an alarm when an intrusion is detected. Section 9.1.6 of the TSS states that the system responds to potential threats by sending an alert, which can be configured by users of the system. IDS_RDR.1.1 states that the System shall provide Administrators and/or Monitors with the capability to read System data. Section 9.1.6 of the TSS shows that all users can view all data in the system. IDS_RDR.1.2 states that the System shall provide the System data in a manner suitable for the user to interpret the information. Section 9.1.6 of the TSS verifies this. IDS_RDR.1.3 states that the system only allows authorized users read access.

IDS_STG.1.1 and IDS_STG.1.2 state that the System shall protect the stored System data from unauthorized deletion and modification. Section 9.1.6 of the TSS shows that data is protected from unauthorized deletion or modification. IDS_STG.1.3 and IDS_STG.2.1 states that the System shall ensure that System data will be maintained when storage is full or system failure, as well as requirements for how new system data is handled when storage is full. Section 9.1.6 of the TSS details that logs are archived when they get large enough and only authorized users with special rights can add data when storage is full.

10 Security Problem Definition Rationale

10.1 Security Objectives Rationale

The following table provides a mapping with rationale to identify the security objectives that address the stated assumptions and threats.

Assumption	Objective	Rationale
A.ACCESS The TOE has access to all the IT System data it needs to perform its functions.	OE.INTROP The TOE is interoperable with the IT System it monitors.	The OE.INTROP objective ensures the TOE has the needed access.
A.DYNMIC The TOE will be managed in a manner that allows it to appropriately address changes in the IT System the TOE monitors.	OE.INTROP The TOE is interoperable with the IT System it monitors.	The OE.INTROP objective ensures the TOE has the proper access to the IT System.
	OE.PERSON Personnel working as authorized administrators shall be carefully selected and trained for proper operation of the System.	The OE.PERSON objective ensures that the TOE will be managed appropriately.
A.ASCOPE The TOE is appropriately scalable to the IT System the TOE monitors.	OE.INTROP The TOE is interoperable with the IT System it monitors.	The OE.INTROP objective ensures the TOE has the necessary interactions with the IT System it monitors.
A.PROTCT The TOE hardware and software critical to security policy enforcement will be protected from unauthorized physical modification.	OE. PHYCAL Those responsible for the TOE must ensure that those parts of the TOE critical to security policy are protected from any physical attack.	The OE.PHYCAL provides for the physical protection of the TOE hardware and software.
A.LOCATE The processing resources of the TOE will be located within controlled access facilities, which will prevent unauthorized physical access.	OE. PHYCAL Those responsible for the TOE must ensure that those parts of the TOE critical to security policy are protected from any physical attack.	The OE.PHYCAL provides for the physical protection of the TOE.
A.MANAGE There will be one or more competent individuals assigned to manage the TOE and the security of the information it contains.	OE.PERSON Personnel working as authorized administrators shall be carefully selected and trained for proper operation of the System.	The OE.PERSON objective ensures all authorized administrators are qualified and trained to manage the TOE.
A.NOEVIL The authorized administrators are not careless, willfully negligent, or hostile, and will follow and abide by the instructions provided by the TOE documentation.	OE.INSTAL Those responsible for the TOE must ensure that the TOE is delivered, installed, managed, and operated in a manner which is consistent with IT security.	The OE.INSTAL objective ensures that the TOE is properly installed and operated.
	OE. PHYCAL Those responsible for the TOE must ensure that those parts of the TOE critical to security policy are protected from any physical attack.	The OE.PHYCAL objective provides for physical protection of the TOE by authorized administrators.
	OE.CREDEN Those responsible for the TOE must ensure that all access credentials are protected by the users in a manner which is consistent with IT security.	The OE.CREDEN objective supports this assumption by requiring protection of all authentication data.

A.NOTRST The TOE can only be accessed by authorized users.	OE. PHYCAL Those responsible for the TOE must ensure that those parts of the TOE critical to security policy are protected from any physical attack.	The OE.PHYCAL objective provides for physical protection of the TOE to protect against unauthorized access.
	OE.CREDEN Those responsible for the TOE must ensure that all access credentials are protected by the users in a manner which is consistent with IT security.	The OE.CREDEN objective supports this assumption by requiring protection of all authentication data.
A.GENPUR There are no general-purpose computing capabilities (e.g., the ability to execute arbitrary code or applications) on the TOE.	OE.GENPUR There are no general-purpose computing capabilities (e.g., the ability to execute arbitrary code or applications) on the TOE.	The OE.GENPUR objective supports this assumption by stating the TOE will not be used for any purposes other than those prescribed by the vendor. Specifically, the addition of non-TOE software or firmware that would add non-TSF functionality or present additional threat vectors is prohibited.
A.UPDATE The TOE will connect to the MAX Network for signature updates and to upload detected malware.	OE.UPDATE The Operational Environment receives detected malware from the TOE and provides signature updates from the TOE on a regular basis.	The OE.UPDATE objective supports this assumption by stating the environment receives detected malware from the TOE. The environment also provides signature updates from the TOE on a regular basis.

Table 10-1: Assumption to Objective Mapping

Threat	Objective	Rationale
T.COMINT An unauthorized user may attempt to compromise the integrity of the data collected and produced by the TOE by bypassing a security mechanism.	O.IDAUTH The TOE must be able to identify and authenticate users prior to allowing access to TOE functions and data.	The O.IDAUTH objective provides for authentication of users prior to any TOE data access (FAU_SAR.2, FAU_STG.2, FIA_UAU.1, FIA_AFL.1 (1), FIA_AFL.1,, FIA_ATD.1, FIA_UID.1, FMT_MOF.1, FMT_MTD.1, FMT_SMR.1, ADV_ARC.1, IDS_RDR.1, IDS_STG.1).
	O.ACCESS The TOE must allow authorized users to access only appropriate TOE functions and data.	The O.ACCESS objective builds upon the O.IDAUTH objective by only permitting authorized users to access TOE data (FAU_SAR.2, FAU_STG.2, FIA_UAU.1, FIA_UID.1, FMT_MOF.1, FMT_MTD.1, FMT_SMF.1 (1), FMT_SMF.1 (2), IDS_RDR.1, IDS_STG.1).
	O.INTEGR The TOE must ensure the integrity of all audit and System data.	The O.INTEGR objective ensures no TOE data will be modified (FAU_STG.2, FMT_MTD.1, ADV_ARC.1, IDS_STG.1).
	O.PROTCT The TOE must protect itself from unauthorized modifications and access to its functions and data.	The O.PROTCT objective addresses this threat by providing TOE self-protection (FAU_STG.2, FMT_MOF.1, FMT_MTD.1, FMT_SMF.1 (1), FMT_SMF.1 (2), ADV_ARC.1, IDS_STD.1).
T.COMDIS An unauthorized user may attempt to disclose the data collected and produced by the TOE by bypassing a security mechanism.	O.IDAUTH The TOE must be able to identify and authenticate users prior to allowing access to TOE functions and data.	The O.IDAUTH objective provides for authentication of users prior to any TOE data access (FAU_SAR.2, FAU_STG.2, FIA_UAU.1, FIA_AFL.1 (1), FIA_AFL.1, FIA_ATD.1, FIA_UID.1, FMT_MOF.1, FMT_MTD.1, FMT_SMR.1, ADV_ARC.1, IDS_RDR.1, IDS_STG.1).
	O.ACCESS The TOE must allow authorized users to access only appropriate TOE functions and data.	The O.ACCESS objective builds upon the O.IDAUTH objective by only permitting authorized users to access TOE data (FAU_SAR.2, FAU_STG.2, FIA_UAU.1, FIA_UID.1, FMT_MOF.1, FMT_MTD.1, FMT_SMF.1 (1), FMT_SMF.1 (2), IDS_RDR.1, IDS_STG.1).

Threat	Objective	Rationale
	O.PROTCT The TOE must protect itself from unauthorized modifications and access to its functions and data.	The O.PROTCT objective addresses this threat by providing TOE self-protection (FAU_STG.2, FMT_MOF.1, FMT_MTD.1, FMT_SMF.1(1), FMT_SMF.1(2), ADV_ARC.1, IDS_STD.1).
	O.EXPORT When any IDS component makes its data available to another IDS components, the TOE will ensure the confidentiality of the System data.	The O.EXPORT objective ensures that confidentiality of TOE data will be maintained (FPT_ITA.1, FPT_ITC.1, FPT_ITI.1).
T.LOSSOF An unauthorized user may attempt to remove or destroy data collected and produced by the TOE.	O.IDAUTH The TOE must be able to identify and authenticate users prior to allowing access to TOE functions and data.	The O.IDAUTH objective provides for authentication of users prior to any TOE data access (FAU_SAR.2, FAU_STG.2, FIA_UAU.1, FIA_ATD.1, FIA_UID.1, FMT_MOF.1, FMT_MTD.1, FMT_SMR.1, ADV_ARC.1, IDS_RDR.1, IDS_STG.1).
	O.ACCESS The TOE must allow authorized users to access only appropriate TOE functions and data.	The O.ACCESS objective builds upon the O.IDAUTH objective by only permitting authorized users to access TOE data (FAU_SAR.2, FAU_STG.2, FIA_UAU.1, FIA_UID.1, FMT_MOF.1, FMT_MTD.1, FMT_SMF.1 (1), FMT_SMF.1 (2), IDS_RDR.1, IDS_STG.1).
	O.INTEGR The TOE must ensure the integrity of all audit and System data.	The O.INTEGR objective ensures no TOE data will be deleted (FAU_STG.2, FMT_MTD.1, ADV_ARC.1, IDS_STG.1).
	O.PROTCT The TOE must protect itself from unauthorized modifications and access to its functions and data.	The O.PROTCT objective addresses this threat by providing TOE self-protection (FAU_STG.2, FMT_MOF.1, FMT_MTD.1, FMT_SMF.1 (1), FMT_SMF.1 (2), ADV_ARC.1, IDS_STD.1).
T.NOHALT An unauthorized user may attempt to compromise the continuity of the System's collection and analysis functions by halting execution of the TOE.	O.IDAUTH The TOE must be able to identify and authenticate users prior to allowing access to TOE functions and data.	The O.IDAUTH objective provides for authentication of users prior to any TOE function accesses (FAU_SAR.2, FAU_STG.2, FIA_UAU.1, FIA_AFL.1 (1), FIA_AFL.1, FIA_ATD.1, FIA_UID.1, FMT_MOF.1, FMT_MTD.1, FMT_SMR.1, ADV_ARC.1, IDS_RDR.1, IDS_STG.1).

Threat	Objective	Rationale
	O.ACCESS The TOE must allow authorized users to access only appropriate TOE functions and data.	The O.ACCESS objective builds upon the O.IDAUTH objective by only permitting authorized users to access TOE functions (FAU_SAR.2, FAU_STG.2, FIA_UAU.1, FIA_UID.1, FMT_MOF.1, FMT_MTD.1, FMT_SMF.1 (1), FMT_SMF.1 (2), IDS_RDR.1, IDS_STG.1).
	O.IDSCAN The Scanner must collect and store static configuration information that might be indicative of the potential for a future intrusion or the occurrence of a past intrusion of an IT System.	The O.IDSCAN, O.IDSENS, and O.IDANLZ objectives address this threat by requiring the TOE to collect and analyze System data, which includes attempts to halt the TOE.
	O.IDSENS The Sensor must collect and store information about all events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets and the IDS.	The O.IDSCAN, O.IDSENS, and O.IDANLZ objectives address this threat by requiring the TOE to collect and analyze System data, which includes attempts to halt the TOE.
	O.IDANLZ The Analyzer must accept data from IDS Sensors or IDS Scanners and then apply analytical processes and information to derive conclusions about intrusions (past, present, or future).	The O.IDSCAN, O.IDSENS, and O.IDANLZ objectives address this threat by requiring the TOE to collect and analyze System data, which includes attempts to halt the TOE.
T.PRIVIL An unauthorized user may gain access to the TOE and exploit system privileges to gain access to TOE security functions and data.	O.IDAUTH The TOE must be able to identify and authenticate users prior to allowing access to TOE functions and data.	The O.IDAUTH objective provides for authentication of users prior to any TOE function accesses (FAU_SAR.2, FAU_STG.2, FIA_UAU.1, FIA_AFL.1 (1), FIA_AFL.1, FIA_ATD.1, FIA_UID.1, FMT_MOF.1, FMT_MTD.1, FMT_SMR.1, ADV_ARC.1, IDS_RDR.1, IDS_STG.1).
	O.ACCESS The TOE must allow authorized users to access only appropriate TOE functions and data.	The O.ACCESS objective builds upon the O.IDAUTH objective by only permitting authorized users to access TOE functions (FAU_SAR.2, FAU_STG.2, FIA_UAU.1, FIA_UID.1, FMT_MOF.1, FMT_MTD.1, FMT_SMF.1 (1), FMT_SMF.1 (2), IDS_RDR.1, IDS_STG.1).

Threat	Objective	Rationale
	O.PROTCT The TOE must protect itself from unauthorized modifications and access to its functions and data.	The O.PROTCT objective addresses this threat by providing TOE self-protection (FAU_STG.2, FMT_MOF.1, FMT_MTD.1, FMT_SMF.1 (1), FMT_SMF.1 (2), ADV_ARC.1, IDS_STD.1).
T.IMPCON An unauthorized user may inappropriately change the configuration of the TOE causing potential intrusions to go undetected.	OE.INSTAL Those responsible for the TOE must ensure that the TOE is delivered, installed, managed, and operated in a manner which is consistent with IT security.	The OE.INSTAL objective states the authorized Administrators will configure the TOE properly.
	O.EADMIN The TOE must include a set of functions that allow effective management of its functions and data.	The O.EADMIN objective ensures the TOE has all the necessary Administrator functions to manage the product (FAU_SAR.1, FAU_SAR.3, FAU_SEL.1, FMT_SMF.1 (1), FMT_SMF.1 (2), ADV_ARC.1, IDS_RDR.1).
	O.IDAUTH The TOE must be able to identify and authenticate users prior to allowing access to TOE functions and data.	The O.IDAUTH objective provides for authentication of users prior to any TOE function accesses (FAU_SAR.2, FAU_STG.2, FIA_UAU.1, FIA_ATD.1, FIA_UID.1, FMT_MOF.1, FMT_MTD.1, FMT_SMR.1, ADV_ARC.1, IDS_RDR.1, IDS_STG.1).
	O.ACCESS The TOE must allow authorized users to access only appropriate TOE functions and data.	The O.ACCESS objective builds upon the O.IDAUTH objective by only permitting authorized users to access TOE functions (FAU_SAR.2, FAU_STG.2, FIA_UAU.1, FIA_UID.1, FMT_MOF.1, FMT_MTD.1, FMT_SMF.1 (1), FMT_SMF.1 (2), IDS_RDR.1, IDS_STG.1).
T.INFLUX An unauthorized user may cause malfunction of the TOE by creating an influx of data that the TOE cannot handle.	O.OFLOWS The TOE must appropriately handle potential audit and System data storage overflows.	The O.OFLOWS objective counters this threat by requiring the TOE handle data storage overflows (FAU_STG.2, FAU_STG.4, IDS_STG.1, IDS_STG.2).
T.FACCNT Unauthorized attempts to access TOE data or security functions may go undetected.	O.AUDITS The TOE must record audit records for data accesses and use of the System functions.	The O.AUDITS objective counters this threat by requiring the TOE to audit attempts for data accesses and use of TOE functions (FAU_GEN.1, FAU_SEL.1, FAU_STG.2,

Threat	Objective	Rationale
		FMT_SMF.1 (1), FMT_SMF.1 (2), ADV_ARC.1, FPT_STM.1).
T.SCNCFG Improper security configuration settings may exist in the IT System the TOE monitors.	O.IDSCAN The Scanner must collect and store static configuration information that might be indicative of the potential for a future intrusion or the occurrence of a past intrusion of an IT System.	The O.IDSCAN objective counters this threat by requiring a TOE that contains a Scanner to collect and store static configuration information that might be indicative of a configuration setting change. The ST will state whether this threat must be addressed by a Scanner (IDS_SDC.1).
T.SCNMLC Users could execute malicious code on an IT System that the TOE monitors which causes modification of the IT System protected data or undermines the IT System security functions.	O.IDSCAN The Scanner must collect and store static configuration information that might be indicative of the potential for a future intrusion or the occurrence of a past intrusion of an IT System.	The O.IDSCAN objective counters this threat by requiring a TOE that contains a Scanner to collect and store static configuration information that might be indicative of malicious code. The ST will state whether this threat must be addressed by a Scanner (IDS_SDC.1).
T.SCNVUL Vulnerabilities may exist in the IT System the TOE monitors.	O.IDSCAN The Scanner must collect and store static configuration information that might be indicative of the potential for a future intrusion or the occurrence of a past intrusion of an IT System.	The O.IDSCAN objective counters this threat by requiring a TOE that contains a Scanner to collect and store static configuration information that might be indicative of a vulnerability. The ST will state whether this threat must be addressed by a Scanner (IDS_SDC.1).
T.FALACT The TOE may fail to react to identified or suspected vulnerabilities or inappropriate activity.	O.RESPON The TOE must respond appropriately to analytical conclusions.	The O.RESPON objective ensures the TOE reacts to analytical conclusions about suspected vulnerabilities or inappropriate activity (IDS_RCT.1).
T.FALREC The TOE may fail to recognize vulnerabilities or inappropriate activity based on IDS data received from each data source.	O.IDANLZ The Analyzer must accept data from IDS Sensors or IDS Scanners and then apply analytical processes and information to derive conclusions about intrusions (past, present, or future).	The O.IDANLZ objective provides the function that the TOE will recognize vulnerabilities or inappropriate activity from a data source (IDS_ANL.1).
T.FALASC The TOE may fail to identify vulnerabilities or inappropriate activity based on association of IDS data received from all data sources.	O.IDANLZ The Analyzer must accept data from IDS Sensors or IDS Scanners and then apply analytical processes and information to derive conclusions about intrusions (past, present, or future).	The O. IDANLZ objective provides the function that the TOE will recognize vulnerabilities or inappropriate activity from multiple data sources (IDS_ANL.1).

Threat	Objective	Rationale
T.MISUSE Unauthorized accesses and activity indicative of misuse may occur on an IT System the TOE monitors.	O.AUDITS The TOE must record audit records for data accesses and use of the System functions.	The O.AUDITS (FAU_GEN.1, FAU_SEL.1, FAU_STG.2, FMT_SMF.1 (1), FMT_SMF.1 (2), ADV_ARC.1, and FPT_STM.1) and O.IDSENS (IDS_SDC.1) objectives address this threat by requiring a TOE that contains a Sensor to collect audit and Sensor data.
	O.IDSENS The Sensor must collect and store information about all events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets and the IDS.	The O.AUDITS (FAU_GEN.1, FAU_SEL.1, FAU_STG.2, FMT_SMF.1 (1), FMT_SMF.1 (2), ADV_ARC.1, FPT_STM.1) and O.IDSENS (IDS_SDC.1) objectives address this threat by requiring a TOE that contains a Sensor to collect audit and Sensor data.
T.INADVE Inadvertent activity and access may occur on an IT System the TOE monitors.	O.AUDITS The TOE must record audit records for data accesses and use of the System functions.	The O.AUDITS (FAU_GEN.1, FAU_SEL.1, FAU_STG.2, FMT_SMF.1 (1), FMT_SMF.1 (2), ADV_ARC.1, and FPT_STM.1) and O.IDSENS (IDS_SDC.1) objectives address this threat by requiring a TOE that contains a Sensor to collect audit and Sensor data.
	O.IDSENS The Sensor must collect and store information about all events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets and the IDS.	The O.AUDITS (FAU_GEN.1, FAU_SEL.1, FAU_STG.2, FMT_SMF.1 (1), FMT_SMF.1 (2), ADV_ARC.1, and FPT_STM.1) and O.IDSENS (IDS_SDC.1) objectives address this threat by requiring a TOE that contains a Sensor to collect audit and Sensor data.
T.MISACT Malicious activity, such as introductions of Trojan horses and viruses, may occur on an IT System the TOE monitors.	O.AUDITS The TOE must record audit records for data accesses and use of the System functions.	The O.AUDITS (FAU_GEN.1, FAU_SEL.1, FAU_STG.2, FMT_SMF.1 (1), FMT_SMF.1 (2), ADV_ARC.1, and FPT_STM.1) and O.IDSENS (IDS_SDC.1) objectives address this threat by requiring a TOE that contains a Sensor to collect audit and Sensor data.
	O.IDSENS The Sensor must collect and store information about all events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets and the IDS.	The O.AUDITS (FAU_GEN.1, FAU_SEL.1, FAU_STG.2, FMT_SMF.1 (1), FMT_SMF.1 (2), ADV_ARC.1, and FPT_STM.1) and O.IDSENS (IDS_SDC.1) objectives address this threat by requiring a TOE that contains a Sensor to

Threat	Objective	Rationale
		collect audit and Sensor data.
T.EAVESDROPPING A malicious user could eavesdrop on network traffic to gain unauthorized access to TOE data.	O.EAVESDROPPING The TOE will encrypt TSF data that traverses the network to prevent malicious users from gaining unauthorized access to TOE data.	O.EAVESDROPPING (FCS_CKM.1(1), FCS_CKM.1(2) FCS_CKM.4, FCS_COP.1(1), FCS_COP.1(2), FPT_TRP.1, FPT_ITC.1, FPT_ITI.1,) mitigates T.EAVESDROPPING by ensuring that all communication to/from the TOE is not sent unless it is encrypted.

Table 10-2: Threat to Objective Mapping

10.2 Operational Security Policy Rationale

The following section details each Operational Security Policy and which objectives map to them, along with rationale for why each one does:

OSP	Objective	Rationale
P.DETECT Static configuration information that might be indicative of the potential for a future intrusion or the occurrence of a past intrusion of an IT System or events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets must be collected.	O.IDSCAN The Scanner must collect and store static configuration information that might be indicative of the potential for a future intrusion or the occurrence of a past intrusion of an IT System.	The O.IDSCAN objective addresses this policy by requiring collection of Scanner data (IDS_SDC.1).
	O.IDSENS The Sensor must collect and store information about all events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets and the IDS.	The O.IDSENS objective addresses this policy by requiring collection of Sensor data (IDS_SDC.1).
	O.AUDITS The TOE must record audit records for data accesses and use of the System functions.	The O.AUDITS objective addresses this policy by requiring collection of audit data (FAU_GEN.1, FAU_SEL.1, FAU_STG.4, ADV_ARC.1, FMT_SMF.1 (1), FMT_SMF.1 (2), and FPT_STM.1).
	OE.TIME The IT Environment will provide reliable time stamp to the TOE. Time stamps associated with an audit record must be reliable.	The OE.TIME objective addresses this policy by providing reliable time stamps to the TOE (FPT_STM.1).
P.ANALYZ Analytical processes and information to derive conclusions about intrusions (past,	O.IDANLZ The Analyzer must accept data from IDS Sensors or IDS Scanners and then apply	The O.IDANLZ objective requires analytical processes be applied to data collected from

present, or future) must be applied to IDS data and appropriate response actions taken.	analytical processes and information to derive conclusions about intrusions (past, present, or future).	Sensors and Scanners (IDS_ANL.1).
P.MANAGE The TOE shall only be managed by authorized users.	O.PROTCT The TOE must protect itself from unauthorized modifications and access to its functions and data.	The O.PROTCT objective addresses this policy by providing TOE self-protection (FAU_STG.2, FMT_MOF.1, FMT_SMF.1 (1), FMT_SMF.1 (2), FMT_MTD.1, ADV_ARC.1, and IDS_STG.1).
	O.EADMIN The TOE must include a set of functions that allow effective management of its functions and data.	The O.EADMIN objective ensures there is a set of functions for Administrators to use (FAU_SAR.1, FAU_SAR.2, FAU_SEL.1, FMT_SMF.1 (1), FMT_SMF.1 (2), ADV_ARC.1, and IDS_RCT.1).
	O.ACCESS The TOE must allow authorized users to access only appropriate TOE functions and data.	The O.ACCESS objective builds upon the O.IDAUTH objective by only permitting authorized users to access TOE functions (FAU_SAR.2, FAU_STG.2, FIA_UAU.1, FIA_UID.1, FMT_MOF.1, FMT_MTD.1, FMT_SMF.1 (1), FMT_SMF.1 (2), IDS_RDR.1, and IDS_STG.1).
	O.IDAUTH The TOE must be able to identify and authenticate users prior to allowing access to TOE functions and data.	The O.IDAUTH objective provides for authentication of users prior to any TOE function accesses (FAU_SAR.2, FAU_STG.2, FIA_UAU.1, FIA_AFL.1 (1), FIA_AFL.1, FIA_ATD.1, FIA_UID.1, FMT_MOF.1, FMT_MTD.1, FMT_SMR.1, ADV_ARC.1, IDS_RDR.1, and IDS_STG.1).
	OE.INSTAL Those responsible for the TOE must ensure that the TOE is delivered, installed, managed, and operated in a manner which is consistent with IT security.	The OE.INSTAL objective supports the OE.PERSON objective by ensuring Administrator follow all provided documentation and maintain the security policy.
	OE.CREDEN Those responsible for the TOE must ensure that all access credentials are protected by the users in a manner which is consistent with IT security.	The OE.CREDEN objective requires Administrators to protect all authentication data.

	OE.PERSON Personnel working as authorized administrators shall be carefully selected and trained for proper operation of the System.	The OE.PERSON objective ensures competent Administrators will manage the TOE.
P.ACCESS All data collected and produced by the TOE shall only be used for authorized purposes.	O.PROTCT The TOE must protect itself from unauthorized modifications and access to its functions and data.	The O.PROTCT objective addresses this policy by providing TOE self-protection (FAU_STG.2, FMT_MOF.1, FMT_MTD.1, FMT_SMF.1 (1), FMT_SMF.1 (2), ADV_ARC.1, and IDS_STG.1).
	O.ACCESS The TOE must allow authorized users to access only appropriate TOE functions and data.	The O.ACCESS objective builds upon the O.IDAUTH objective by only permitting authorized users to access TOE functions (FAU_STG.2, FMT_MOF.1, FMT_MTD.1, FMT_SMF.1 (1), FMT_SMF.1 (2), ADV_ARC.1, and IDS_STG.1).
	O.IDAUTH The TOE must be able to identify and authenticate users prior to allowing access to TOE functions and data.	The O.IDAUTH objective provides for authentication of users prior to any TOE function accesses (FAU_SAR.2, FAU_STG.2, FIA_AFL.1 (1), FIA_AFL.1, FIA_UAU.1, FIA_ATD.1, FIA_UID.1, FMT_MOF.1, FMT_MTD.1, FMT_SMR.1, ADV_ARC.1, IDS_RDR.1, and IDS_STG.1).
	OE.AUDIT_PROTECTION The IT Environment will provide the capability to protect audit information.	The OE.AUDIT_PROTECTION objective addresses this policy by providing audit information protection (FAU_STG.2).
P.ACCACT Users of the TOE shall be accountable for their actions within the IDS.	O.IDAUTH The TOE must be able to identify and authenticate users prior to allowing access to TOE functions and data.	The O.IDAUTH objective supports this objective by ensuring each user is uniquely identified and authenticated (FAU_SAR.2, FAU_STG.2, FIA_UAU.1, FIA_AFL.1 (1), FIA_AFL.1, FIA_ATD.1, FIA_UID.1, FMT_MOF.1, FMT_MTD.1, FMT_SMR.1, ADV_ARC.1, IDS_RDR.1, and IDS_STG.1).
	O.AUDITS The TOE must record audit records for data accesses and use of the System functions.	The O.AUDITS objective implements this policy by requiring auditing of all data accesses and use of TOE functions (FAU_SAR.2, FAU_STG.2, FIA_UAU.1, FIA_ATD.1, FIA_UID.1,

		FMT_MOF.1, FMT_MTD.1, FMT_SMF.1 (1), FMT_SMF.1 (2), FMT_SMR.1, ADV_ARC.1, IDS_RDR.1, and IDS_STG.1).
	OE.TIME The IT Environment will provide reliable timestamps to the TOE.	The OE.TIME objective addresses this policy by providing reliable time stamps to the TOE (FPT_STM.1).
	OE.AUDIT_SORT The IT Environment will provide the capability to sort the audit information.	The OE.AUDIT_SORT objective addresses this policy by providing audit sorting capabilities (FAU_SAR.3).
P.INTGTY Data collected and produced by the TOE shall be protected from modification.	O.INTEGR The TOE must ensure the integrity of all audit and System data.	The O.INTEGR objective ensures the protection of data from modification (FAU_STG.2, FMT_MTD.1, FPT_ITC.1, FPT_ITI.1, ADV_ARC.1, and IDS_STG.1).
P.PROTCT The TOE shall be protected from unauthorized accesses and disruptions of TOE data and functions.	O.OFLOWS The TOE must appropriately handle potential audit and System data storage overflows.	The O.OFLOWS objective counters this policy by requiring the TOE handle disruptions (FAU_STG.2, FAU_STG.4, IDS_STG.1, and IDS_STG.2).
	OE. PHYCAL Those responsible for the TOE must ensure that those parts of the TOE critical to security policy are protected from any physical attack.	The OE.PHYCAL objective protects the TOE from unauthorized physical modifications.

Table 10-3: OSP to Objective Mapping

10.3 Security Functional Requirements Rationale

The following table provides a mapping with rationale to identify the security functional requirement components that address the stated TOE and environment objectives.

Objective	Security Functional Components	Rationale
O.PROTCT The TOE must protect itself from unauthorized modifications and access to its functions and data	FAU_STG.2 Guarantees of Audit Data Availability	The TOE is required to protect the audit data from deletion as well as guarantee the availability of the audit data in the event of storage exhaustion, failure or attack [FAU_STG.2]. The System is required to protect the System data from any modification and unauthorized deletion, as well as guarantee the availability of the data in the event of storage exhaustion, failure or attack [IDS_STG.1]. The TOE is
	FMT_MOF.1 Management of Security Functions Behavior	
	FMT_MTD.1 Management of TSF Data	
	FMT_SMF.1(1) Specification of management functions	
	FMT_SMF.1(2) Specification of management functions	

Objective	Security Functional Components	Rationale
	ADV_ARC.1 Architectural Design IDS_STD.1 Guarantee of System Data Availability	required to provide the ability to restrict managing the behavior of functions of the TOE to authorized users of the TOE [FMT_MOF.1]. Only authorized Administrators of the System may query and add System and audit data, and authorized Administrators of the TOE may query and modify all other TOE data [FMT_MTD.1]. The TOE must perform security management and IDS management for auditing, attributes, reports, alerts, and defense perspective data [FMT_SMF.1 (1), FMT_SMF.1 (2)]. The TOE must ensure that all functions are invoked and succeed before each function may proceed [ADV_ARC.1]. The TSF must be protected from interference that would prevent it from performing its functions [ADV_ARC.1].
O.IDSCAN The Scanner must collect and store static configuration information that might be indicative of the potential for a future intrusion or the occurrence of a past intrusion of an IT System	IDS_SDC.1 System Data Collection	A System containing a Scanner is required to collect and store static configuration information of an IT System. The type of configuration information collected must be defined in the ST [IDS_SDC.1]
O.EXPORT When any IDS component makes its data available to another IDS components, the TOE will ensure the confidentiality of the System data.	FPT_ITA.1 Inter-TSF availability within a defined availability metric FPT_ITC.1 Inter-TSF confidentiality during transmission FPT_ITI.1 Inter-TSF detection of modification	The TOE must make the collected data available to other IT products [FPT_ITA.1]. The TOE must protect all data from modification and ensure its integrity when the data is transmitted to another IT product and between TOE instances [FPT_ITC.1, FPT_ITI.1].
O.IDSENS The Sensor must collect and store information about all events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets and the IDS	IDS_SDC.1 System Data Collection	A System containing a Sensor is required to collect events indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets of an IT System. These events must be defined in the ST [IDS_SDC.1].

Objective	Security Functional Components	Rationale
O.IDANLZ The Analyzer must accept data from IDS Sensors or IDS Scanners and then apply analytical processes and information to derive conclusions about intrusions (past, present, or future).	IDS_ANL.1 Analyzer Analysis	The Analyzer is required to perform intrusion analysis and generate conclusions [IDS_ANL.1].
O.RESPON The TOE must respond appropriately to analytical conclusions	IDS_RCT.1 Analyzer React	The TOE is required to respond accordingly in the event an intrusion is detected [IDS_RCT.1].
O.EADMIN The TOE must include a set of functions that allow effective management of its functions and data	FAU_SAR.1 Audit Review FAU_SAR.3 Selectable Audit Review FAU_SEL.1 Selective Audit ADV_ARC.1 Architectural Design IDS_RDR.1 Restricted Data Review FMT_SMF.1(1) Specification of Management Functions FMT_SMF.1(2) Specification of Management Functions	The TOE must provide the ability to review and manage the audit trail of the System [FAU_SAR.1, FAU_SEL.1]. The System must provide the ability for authorized Administrators to view all System data collected and produced [IDS_RDR.1]. The TOE must perform security management and IDS management for auditing, attributes, reports, alerts, and defense perspective data [FMT_SMF.1 (1), FMT_SMF.1 (2)]. The TOE must ensure that all functions are invoked and succeed before each function may proceed [ADV_ARC.1]. The TSF must be protected from interference that would prevent it from performing its functions [ADV_ARC.1].
O.ACCESS The TOE must allow authorized users to access only appropriate TOE functions and data.	FAU_SAR.2 Restricted Audit Review FAU_STG.2 Guarantees of Audit Data Availability FIA_UAU.1 Timing of Authentication FIA_UID.1 Timing of Identification FMT_MOF.1 Management of Security Functions Behavior FMT_MTD.1 Management of TSF Data FMT_SMF.1(1) Specification of management functions FMT_SMF.1(2) Specification of management functions IDS_RDR.1 Restricted Data Review	The TOE is required to restrict the review of audit data to those granted with explicit read-access [FAU_SAR.2]. The System is required to restrict the review of System data to those granted with explicit read-access [IDS_RDR.1]. The TOE is required to protect the audit data from deletion as well as guarantee the availability of the audit data in the event of storage exhaustion, failure or attack [FAU_STG.2]. The System is required to protect the System data from any modification and unauthorized deletion [IDS_STG.1]. Users authorized to access the TOE are defined using an identification and authentication process [FIA_UID.1,

Objective	Security Functional Components	Rationale
	IDS_STG.1 Guarantee of System Data Availability	FIA_UAU.1]. The TOE is required to provide the ability to restrict managing the behavior of functions of the TOE to authorized users of the TOE [FMT_MOF.1]. The TOE must perform security management and IDS management for auditing, attributes, reports, alerts, and defense perspective data [FMT_SMF.1 (1), FMT_SMF.1 (2)]. Only authorized Administrators of the System may query and add System and audit data, and authorized Administrators of the TOE may query and modify all other TOE data [FMT_MTD.1].
O.IDAUTH The TOE must be able to identify and authenticate users prior to allowing access to TOE functions and data.	FAU_SAR.2 Restricted Audit Review FAU_STG.2 Guarantees of Audit Data Availability FIA_UAU.1 Timing of Authentication FIA_AFL.1(1) Authentication Failure Handling FIA_ATD.1 User Attribute Definition FIA_UID.1 Timing of Identification FMT_MOF.1 Management of Security Functions Behavior FMT_MTD.1 Management of TSF Data FMT_SMR.1 Security Roles ADV_ARC.1 Architectural Design IDS_RDR.1 Restricted Data Review	The TOE is required to restrict the review of audit data to those granted with explicit read-access [FAU_SAR.2]. The System is required to restrict the review of System data to those granted with explicit read-access [IDS_RDR.1]. The TOE is required to protect the stored audit records from unauthorized deletion [FAU_STG.2]. The System is required to protect the System data from any modification and unauthorized deletion, as well as guarantee the availability of the data in the event of storage exhaustion, failure or attack [IDS_STG.1]. Security attributes of subjects use to enforce the authentication policy of the TOE must be defined [FIA_ATD.1]. Users authorized to access the TOE are defined using an identification and authentication process [FIA_UID.1,

Objective	Security Functional Components	Rationale
	IDS_STG.1 Guarantee of System Data Availability	FIA_UAU.1]. Once users attempt to login 5 times unsuccessfully, they are locked out for 30 minutes (FIA_AFL.1 (1). The TOE is required to detect failed authentication attempts of IT products and require administrator action before the IT product can authenticate when a threshold is reached [FIA_AFL.1]. The TOE is required to provide the ability to restrict managing the behavior of functions of the TOE to authorized users of the TOE [FMT_MOF.1]. Only authorized Administrators of the System may query and add System and audit data, and authorized Administrators of the TOE may query and modify all other TOE data [FMT_MTD.1]. The TOE must be able to recognize the different administrative and user roles that exist for the TOE [FMT_SMR.1]. The TOE must ensure that all functions are invoked and succeed before each function may proceed [ADV_ARC.1]. The TSF must be protected from interference that would prevent it from performing its functions [ADV_ARC.1].
O.OFLOWS The TOE must appropriately handle potential audit and System data storage overflows	FAU_STG.2 Guarantee of Audit Data Availability FAU_STG.4 Prevention of Data Loss IDS_STG.1 Guarantee of System Data Availability IDS_STG.2 Prevention of System Data Loss	The TOE is required to protect the audit data from deletion as well as guarantee the availability of the audit data in the event of storage exhaustion, failure or attack [FAU_STG.2]. The TOE must prevent the loss of audit data in the event that its audit trail is full [FAU_STG.4]. The System is required to protect the System data from any modification and unauthorized deletion, as well as guarantee the availability of the data in the event of storage exhaustion, failure or attack [IDS_STG.1]. The System must prevent the loss of audit data in the event that its audit trail is full [IDS_STG.2].
O.AUDITS The TOE must record audit records for data accesses and use of the System functions	FAU_GEN.1 Audit Data Generation FAU_SEL.1 Selective Audit FAU_STG.4 Prevention of Data Loss FMT_SMF.1(1) Specification of management functions	Security-relevant events must be defined and auditable for the TOE [FAU_GEN.1]. The TOE must provide the capability to select which security-relevant events to audit [FAU.SEL.1]. The TOE must prevent the loss of collected data in the event

Objective	Security Functional Components	Rationale
	FMT_SMF.1(2) Specification of management functions ADV_ARC.1 Architectural Design FPT_STM.1 Reliable Time Stamps	that its audit trail is full [FAU_STG.4]. The TOE must perform security management and IDS management for auditing, attributes, reports, alerts, and defense perspective data [FMT_SMF.1 (1), FMT_SMF.1 (2)]. The TOE must ensure that all functions are invoked and succeed before each function may proceed [ADV_ARC.1]. The TSF must be protected from interference that would prevent it from performing its functions [ADV_ARC.1]. Time stamps associated with an audit record must be reliable [FPT_STM.1]
O.INTEGR The TOE must ensure the integrity of all audit and System data	FAU_STG.2 Guarantee of Audit Data Availability FMT_MTD.1 Management of TSF Data ADV_ARC.1 Architectural Design IDS_STG.1 Guarantee of System Data Availability FPT_ITC.1 Inter-TSF confidentiality during transmission FPT_ITI.1 Inter-TSF detection of modification	The TOE is required to protect the audit data from deletion as well as guarantee the availability of the audit data in the event of storage exhaustion, failure or attack [FAU_STG.2]. The System is required to protect the System data from any modification and unauthorized deletion [IDS_STG.1]. Only authorized Administrators of the System may query or add audit and System data [FMT_MTD.1]. The TOE must ensure that all functions to protect the data are not bypassed [ADV_ARC.1]. The TSF must be protected from interference that would prevent it from performing its functions [ADV_ARC.1]. The TOE must make the collected data available to other IT products [FPT_ITA.1]. The TOE must protect all data from modification and ensure its integrity when the data is transmitted to another IT product and between TOE instances [FPT_ITC.1, FPT_ITI.1].
OE.TIME The IT Environment will provide reliable time stamp to the TOE. Time stamps associated with an audit record must be reliable	FPT_STM.1 Reliable Time Stamps	The IT Environment will provide reliable time stamp to the TOE. Time stamps associated with an audit record must be reliable [FPT_STM.1].
OE.AUDIT_SORT The IT Environment will provide the capability to sort audit information.	FAU_SAR.3 Selectable Audit Review	The IT environment must provide the ability to review and manage the audit trail of the System to include sorting the audit data [FAU_SAR.3,].

Objective	Security Functional Components	Rationale
OE.AUDIT_PROTECTION The IT Environment will provide the capability to protect audit information.	FPT_STM.1 Reliable Time Stamps	The TOE is required to protect the audit data from deletion as well as guarantee the availability of the audit data in the event of storage exhaustion, failure or attack [FAU_STG.2].
O.EAVESDROPPING The TOE will encrypt TSF data that traverses the network to prevent malicious users from gaining unauthorized access to TOE data.	FCS_CKM.1(1) Cryptographic key generation	FCS_CKM.1 (1) states the TSF shall generate 128 bit keys using AES for OpenSSL communication to/from the MAX Network and to/from the WebUI.
	FCS_CKM.1(2) Cryptographic key generation	FCS_CKM.1 (2) states the TSF shall generate 256 bit keys using AES for OpenSSH communication to/from the CLI and to/from the CMS.
	FCS_CKM.4 Cryptographic key destruction	FCS_CKM.4 states the TSF shall destroy cryptographic keys by overwrite.
	FCS_COP.1(1) Cryptographic operation	FCS_COP.1 (1) states that the TSF uses 128 bit keys using AES in CBC mode for OpenSSL encryption and decryption to/from the MAX Network and to/from the WebUI.
	FCS_COP.1(2) Cryptographic operation	FCS_COP.1 (2) states that the TSF uses 256 bit keys using AES in CBC mode for OpenSSH encryption and decryption to/from the CLI and to/from the CMS.
	FTP_TRP.1 Trusted Path	FTP_TRP.1 states the TOE shall provide a communication path between the TSF and remote users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from modification or disclosure. The TSF shall allow remote users to initiate communication via the trusted path, and it shall require the use of the trusted path for initial user authentication and management of the TOE.
	FPT_ITC.1 Inter-TSF confidentiality during transmission	FPT_ITC.1 states that the TOE shall protect all TSF data transferred to remote trusted IT products. The TOE uses OpenSSL with 128-bit AES encryption to secure transferred data.
	FPT_ITI.1 Inter-TSF detection of modification	FPT-ITI.1 states that the TOE shall have the capability to detect modification using HMAC-MD5 hashing and be able to take action if modifications are detected.

Table 10-4: Security Functional Requirements Rationale

10.4 EAL2 Justification

The threats that were chosen are consistent with attacker of basic attack potential, therefore EAL2 augmented with ALC_FLR.2 was chosen for this ST. ALC_FLR.2 is not required, but provides additional quality assurance to the product.

10.5 Requirement Dependency Rationale

The IDS System PP does satisfy all the requirement dependencies of the Common Criteria. The table below lists each requirement from the IDS System PP with a dependency and indicates whether the dependent requirement was included. As the table indicates, all dependencies have been met.

Functional Component	Dependency	Included
FAU_GEN.1	FPT_STM.1	YES
FAU_SAR.1	FAU_GEN.1	YES
FAU_SAR.2	FAU_SAR.1	YES
FAU_SAR.3	FAU_SAR.1	YES
FAU_SEL.1	FAU_GEN.1	YES
	FMT_MTD.1	YES
FAU_STG.2	FAU_GEN.1	YES
FAU_STG.4	FAU_STG.2	YES
FIA_UAU.1	FIA_UID.1	YES
FMT_MOF.1	FMT_SMR.1	YES
FMT_MTD.1	FMT_SMR.1	YES
FMT_SMR.1	FIA_UID.1	YES
FCS_CKM.1	FCS_COP.1	YES
	FCS_CKM.4	YES
FCS_CKM.4	FCS_CKM.1	YES
FCS_COP.1	FCS_CKM.1	YES
	FCS_CKM.4	YES

Table 10-5: Requirement Dependencies

10.6 Strength of Function Rationale

The TOE minimum strength of function is SOF-basic. The evaluated TOE is intended to operate in commercial and DoD low robustness environments processing unclassified information. This security function is in turn consistent with the security objectives described in section 5.

10.7 Assurance Measures

The SARs for this evaluation have been chosen because they are consistent with the package claim of EAL2. Augmentations to this claim include ALC_FLR.2. ALC_FLR.2 provides assurance that the TOE is updated in a well-defined manner that is consistent with the development security procedures outlined in ALC_DVS.1.

The following table identifies the SARs for this ST.

Component	Document(s)	Rationale
ADV_ARC.1 Security Architecture Description	TOE Design Specification for FireEye MAS and MPS 2000, 4000, and 7000 Series v.5.0	This document describes the security architecture of the TOE.
ADV_FSP.2 Security-enforcing functional specification	Functional Specification Document for FireEye MAS and MPS 2000, 4000, and 7000 Series v.5.0	This document describes the functional specification of the TOE with complete summary.
ADV_TDS.1 Basic Design	TOE Design Specification for FireEye MAS and MPS 2000, 4000, and 7000 Series v.5.0	This document describes the architectural design of the TOE.
AGD_OPE.1 Operational User Guidance	• FireEye Appliance Operator's Guide Version 5.0 • FireEye CMS Operator's Guide Version 5.0 • FireEye CLI Command Reference Guide Version 5.0	This document describes the operational user guidance for FireEye MAS and MPS 2000, 4000, and 7000 Series v.5.0.
AGD_PRE.1 Preparative Procedures	• FireEye™ Appliance Quick Start Guide • Evaluated Configuration for FireEye MAS and MPS 2000, 4000, and 7000 Series v.5.0	This document describes the preparative procedures that need to be done prior to installing FireEye MAS and MPS 2000, 4000, and 7000 Series v.5.0.
ALC_CMC.2 Use of a CM system	• FireEye Configuration Management Plan v0.4	This document describes the authorization controls for the TOE.
ALC_CMS.2 Parts of the TOE CM coverage	• FireEye Configuration Management Plan v0.4	These documents describe the CM scope of the TOE.
ALC_DEL.1 Delivery Procedures	• FireEye Configuration Management Plan v0.4	This document describes product delivery for FireEye MAS and MPS 2000, 4000, and 7000 Series v.5.0 and a description of all procedures used to ensure objectives are not compromised in the delivery process.
ALC_FLR.2 Flaw reporting procedures	• FireEye Configuration Management Plan v0.4 • PRODUCT REQUIREMENTS DOCUMENT r5.0 v0.7	This document describes the processes taken for flaw remediation for the TOE.
ASE_CCL.1 Conformance Claims	FireEye MAS and MPS 2000, 4000, and 7000 Series with Central Management System v.5.0 Security Target v1.2	This document describes the CC conformance claims made by the TOE.
ASE_ECD.1 Extended Components Definition	FireEye MAS and MPS 2000, 4000, and 7000 Series with Central Management System v.5.0 Security Target v1.2	This document provides a definition for all extended components in the TOE.
ASE_INT.1 Security Target Introduction	FireEye MAS and MPS 2000, 4000, and 7000 Series with Central Management System v.5.0 Security Target v1.2	This document describes the Introduction of the Security Target.

Component	Document(s)	Rationale
ASE_OBJ.2 Security Objectives	FireEye MAS and MPS 2000, 4000, and 7000 Series with Central Management System v.5.0 Security Target v1.2	This document describes all of the security objectives for the TOE.
ASE_REQ.2 Derived Security Requirements	FireEye MAS and MPS 2000, 4000, and 7000 Series with Central Management System v.5.0 Security Target v1.2	This document describes all of the security requirements for the TOE.
ASE_SPD.1 Security Problem Definition	FireEye MAS and MPS 2000, 4000, and 7000 Series with Central Management System v.5.0 Security Target v1.2	This document describes the security problem definition of the Security Target.
ASE_TSS.1 TOE Summary Specification	FireEye MAS and MPS 2000, 4000, and 7000 Series with Central Management System v.5.0 Security Target v1.2	This document describes the TSS section of the Security Target.
ATE_COV.1 Evidence of Coverage	• Independent Testing Plan for FIREEYE, INC. FireEye MAS and MPS 2000, 4000, and 7000 Series v5.0 • CC-FE-Malware Analysis - Test Cases-0604.xls • CC-FE-System-Test-Case-0604.xls • CC-FE-Web Product - Test Plan 0604.doc • Malware-Product-Test-Plan-0604.doc	This document provides an analysis of coverage for the TOE.
ATE_FUN.1 Functional Testing	• CC-FE-Malware Analysis - Test Cases-0604.xls • CC-FE-System-Test-Case-0604.xls • CC-FE-Web Product - Test Plan 0604.doc • Malware-Product-Test-Plan-0604.doc	This document describes the functional tests for the TOE.
ATE_IND.2 Independent Testing - sample	Independent Testing Plan for FIREEYE, INC. FireEye MAS and MPS 2000, 4000, and 7000 Series v5.0	This document describes the independent testing for the TOE.
AVA_VAN.2 Vulnerability Analysis	Vulnerability Analysis for FIREEYE, INC. FireEye MAS and MPS 2000, 4000, and 7000 Series v5.0	This document describes the vulnerability analysis of the TOE.

Table 10-6: Assurance Requirements Evidence