Sourcefire Intrusion Detection System
Security Target

Version 1.4

May 19, 2005

Prepared for:
Sourcefire, Incorporated
9770 Patuxent Woods Drive
Columbia, MD 21046

Prepared By:
Science Applications International Corporation
Common Criteria Testing Laboratory
7125 Columbia Gateway Drive, Suite 300
Columbia, MD 21046

LIST OF TABLES

1. Security Target Introduction

This section identifies the Security Target and Target of Evaluation (TOE) identification, ST conventions, ST conformance claims, and the ST organization. Sourcefire, Inc provides the TOE which includes Sourcefire Network Sensor version 3.2.3 and Management Console version 3.2.3. The Network Sensor is embedded in the following products: the Sourcefire NS 500, NS 1000, NS 2000, NS 2100, and NS 3000 models of Intrusion Detection Sensors. The Management Console is embedded on the MC1000 and MC3000 models of the Sourcefire Management Console.

The Security Target contains the following additional sections:

- TOE Description (Section 2)
- Security Environment (Section 3)
- Security Objectives (Section 4)
- IT Security Requirements (Section 5)
- TOE Summary Specification (Section 6)
- Protection Profile Claims (Section 7)
- Rationale (Section 8)

1.1 Security Target, TOE and CC Identification

ST Title – Sourcefire Intrusion Detection System Security Target

ST Version – Version 1.4

ST Date – May 19, 2005

TOE Identification – The Sourcefire Network Sensor version 3.2.3 software is embedded in the following products: NS 500, NS 1000, NS 2000, NS 2100, and NS 3000 models of Intrusion Detection Sensors. The Management Console version 3.2.3 software is embedded in the following products: MC1000, and MC3000 models of the Sourcefire Management Console.

CC Identification – Common Criteria for Information Technology Security Evaluation, Version 2.1, August 1999, ISO/IEC 15408.

1.2 Conformance Claims

This TOE is conformant to the following CC specifications:

- Common Criteria for Information Technology Security Evaluation Part 2: Security functional requirements, Version 2.1, August 1999, ISO/IEC 15408-2.
  - Part 2 Extended (with IDS_SDC.1, IDS_ANL.1, IDS_RCT.1, IDS_RDR.1, IDS_STG.1, and IDS_STG.2)
- Common Criteria for Information Technology Security Evaluation Part 3: Security assurance requirements, Version 2.1, August 1999, ISO/IEC 15408-3.
  - Part 3 Conformant
  - Evaluation Assurance Level 2 (EAL2)

- Strength of Funtional claim: SOF - Basic

This TOE is conformant to the following Protection Profiles (PPs):

- US Government Intrusion Detection System Protection Profile, Version 1.4, February 4, 2002 (IDSSPP).

1.3 Conventions, Terminology, Acronyms

This section specifies the formatting information used in the Security Target.

1.3.1 Conventions

The following conventions have been applied in this document:

- Security Functional Requirements – Part 2 of the CC defines the approved set of operations that may be applied to functional requirements: iteration, assignment, selection, and refinement.
    - Iteration: allows a component to be used more than once with varying operations. In the ST, iteration is indicated by a letter in parenthesis placed at the end of the component. For example FDP_ACC.1(a) and FDP_ACC.1(b) indicate that the ST includes two iterations of the FDP_ACC.1 requirement, a and b.
    - Assignment: allows the specification of an identified parameter. Assignments are indicated using bold and are surrounded by brackets (e.g., [**assignment**]).
    - Selection: allows the specification of one or more elements from a list. Selections are indicated using bold italics and are surrounded by brackets (e.g., [***selection***]).
    - Refinement: allows the addition of details. Refinements are indicated using bold, for additions, and strike-through, for deletions (e.g., "… **all** objects …" or "… ~~some~~ **big** things …").
    - Note that operations already performed in the corresponding Protection Profile are not identified in this Security Target.
- Explicitly stated Security Functional Requirements (i.e., those not found in Part 2 of the CC) are identified with "**(EXP)**".
- Other sections of the ST – Other sections of the ST use bolding to highlight text of special interest, such as captions.

1.3.2 Acronyms

The acronyms used within this Security Target:

ACM	Access Control Management
AGD	Administrator Guidance Document
CC	Common Criteria
CD-ROM	Compact Disk Read Only Memory
CIDR	Classless Inter-Domain Routing
CM	Control Management
DAC	Discretionary Access Control
DO	Delivery Operation
EAL	Evaluation Assurance Level
FIPS	Federal Information Processing Standards Publication
GB	Gigabyte

HTTP	HyperText Transmission Protocol
HTTPS	HyperText Transmission Protocol, Secure
ICMP	Internet Control Message Protocol
ID	Identifier
IDS	Intrusion Detection System
I/O	Input/Output
NIST	National Institute of Standards and Technology
PGP	Pretty Good privacy
PP	Protection Profile
RPC	Remote Procedure Call
SF	Security Functions
SFIDS	Sourcefire Intrusion Detection System
SFR	Security Functional Requirements
ST	Security Target
TCP/IP	Transmission Control Protocol/Internet Protocol
TOE	Target of Evaluation
TOS	Type of Service
TSF	TOE Security Functions
TSP	TOE Security Policy
TSC	TSF Scope of Control
TTL	Time to Live
UDP	User Datagram Protocol
UI	User Interface
URI	Uniform Resource Identifier

2. TOE Description

The TOE is the Sourcefire Network Sensor version 3.2.3 embedded in the following products available from Sourcefire: NS 500, NS 1000, NS 2000, NS 2100, and NS 3000 models of Intrusion Detection Sensors. The TOE also consists of the Management Console version 3.2.3, embedded in the following products available from Sourcefire: MC1000 and MC3000 models of Sourcefire Management Console. These TOE combinations are hereafter referred to collectively as SFIDS. These IDS products are designed by Sourcefire Incorporated, located at 9770 Patuxent Woods Drive, Columbia, Maryland, 21046.

The TOE consists of a Sourcefire appliance that hosts a Linux operating system (SFLinux version 3.2), which supports applications that provide the intrusion detection and associated security management functions. The hardware that the software operates on provides the support necessary for the software applications to exist as processes and to access necessary disk, memory, and network connection resources.

2.1 Product Type

The TOE includes Intrusion Detection appliances that combine open-source and proprietary technology to create a scalable and flexible IDS. SFIDS is used to monitor incoming (and outgoing) network traffic, generally from outside the firewall. All packets on the monitored network are scanned and then compared against a set of rules to determine whether inappropriate traffic, such as system attacks, is being passed over the network. The system then notifies administrators of these attempts.

2.2 Product Description

The TOE application is based upon open source software with proprietary modifications and a proprietary management interface.

All SFIDS applications execute on Sourcefire appliance and a custom Linux (i.e.g, SFLinux) operating system. Administration is performed via a management application running locally in conjunction with each Network Sensor application or on a dedicated Sourcefire Management Console appliance that can manage multiple sensors from a single location. Each of the different Intrusion Detection Sensor appliance models differs only in their throughput capabilities, ranging from 5Mbps for the 500 series to 1Gbps for the 3000.

The Network Sensor is based on an enhanced “snort” version 2.2. Snort is an open source IDS which was originally created by the founder of Sourcefire. Snort (as modified and included in the TOE) is used to read all the packets on the monitored network, and then analyzes them against the rule set that has been created by the IDS administrators.

Based upon the results of the Snort analysis, the packets are either ignored and dropped (those that do not meet any specified rules) or placed into one of two data stores, the Unified Alert or the Unified Files. These are binary data stores where the packet, the rule event and the header information are stored. Alert events are those that trigger immediate responses, such as sending an email to an administrator.

All management is performed through the management application. This can be done either locally on the Intrusion Detection Sensor appliance, or the management can be offloaded to a completely separate Sourcefire Management Console appliance. The management interface is actually a web-based interface, using HTTPS. Through this interface the administrator can perform all management functions, including creating and implementing new rules, managing user accounts, and reviewing the logs/events/alarms. The logs are sent from the appropriate binary store to the management application where they are translated into human readable data (by another instance of snort) and stored in an embedded database. The rules are stored in a separate database.

The TOE has five different authority levels: Administrator, Rule Builder, Data Analysis (hereafter identified simply as “Rule” and “Data,” respectively), Maintenance, and Restrictive Data. The Administrator role has the ability to perform all functions within the system, including creating and implementing rules, managing accounts and managing and viewing the security and IDS event logs. The Rule role can only create rules, edit existing rules, and implement them. The Data role can only manage the IDS event logs, including the abilities to review and delete corresponding data. The Maintenance role can view system statistics and performance metrics, and perform system maintenance tasks. The Restrictive Data role has similar privileges as the Data role does, the exception being that they can be limited to viewing only certain network addresses. No access is allowed to the system until a user has been authenticated and access to various functions is controlled by providing interfaces only to those functions allowed to the authenticated role.

2.3 Product Features

The TOE implements the following features:

Flexible Rule Creation: SFIDS allows the administrator complete control over the rules that govern the detection of attacks and threats, allowing each organization to customize the IDS for their specific requirements.

Rules-based Detection: SFIDS used a rule-based methodology for detecting both known attacks as well as anomalous behavior, allowing detection of even unknown attacks.

Simple Management Interface: SFIDS provides an easy to use web-based interface for all management functions. This makes remote administration simple while providing a familiar UI.

Turnkey Package: SFIDS is a complete package, providing all functions necessary for a secure IDS. The product can be up and running in minutes with a default set of rules to detect known threats.

Detailed Queries for Information: SFIDS provides a robust query engine for sorting and viewing the volumes of collected information.

Detailed Forensic Information: SFIDS provides an administrator with detailed information about threats and attacks, including the entire payload of all suspected packets.

2.4 Security Environment TOE Boundary

The TOE includes both physical and logical boundaries.

2.4.1 Physical Boundaries

The TOE software is physically installed on a Sourcefire appliance and Linux operating system. As such, its physical interfaces are based on and limited by services provided by the TOE hardware. The TOE software uses process, disk, and memory management services of the TOE hardware to execute and manage itself. The TOE software also uses network services provided by the TOE hardware to access network traffic, including monitoring target networks, communication between the Network Sensor and Management Console applications (and associated appliances), and the web-based Management Console interface.

The IT environment, where the TOE operates, consists of the networks that are to be monitored, a mail server that is configured to receive alerts notifications, and a properly configured web browser, which is used to manage the TOE. Note that the networks being monitored and those that allow communication between the Network Sensor and Management Console and associated user management web browsers are necessarily different. All networks, except those being monitored, are assumed to be protected from unauthorized access.

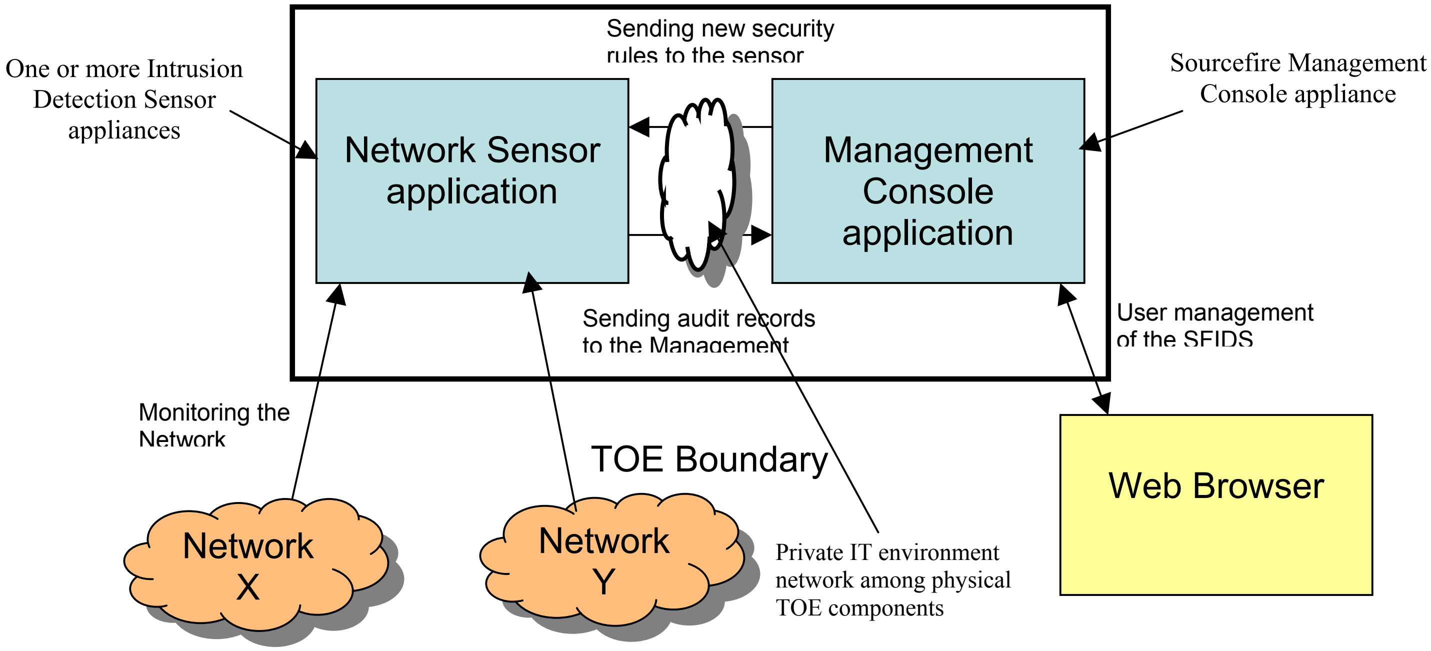
One or more Intrusion Detection Sensor appliances
Sending new security rules to the sensor
Sourcefire Management Console appliance
Network Sensor application
Management Console application
Sending audit records to the Management
User management of the SFIDS
Monitoring the Network
TOE Boundary
Web Browser
Network X
Network Y
Private IT environment network among physical TOE components

Figure 1: SFIDS with Sourcefire Management Console

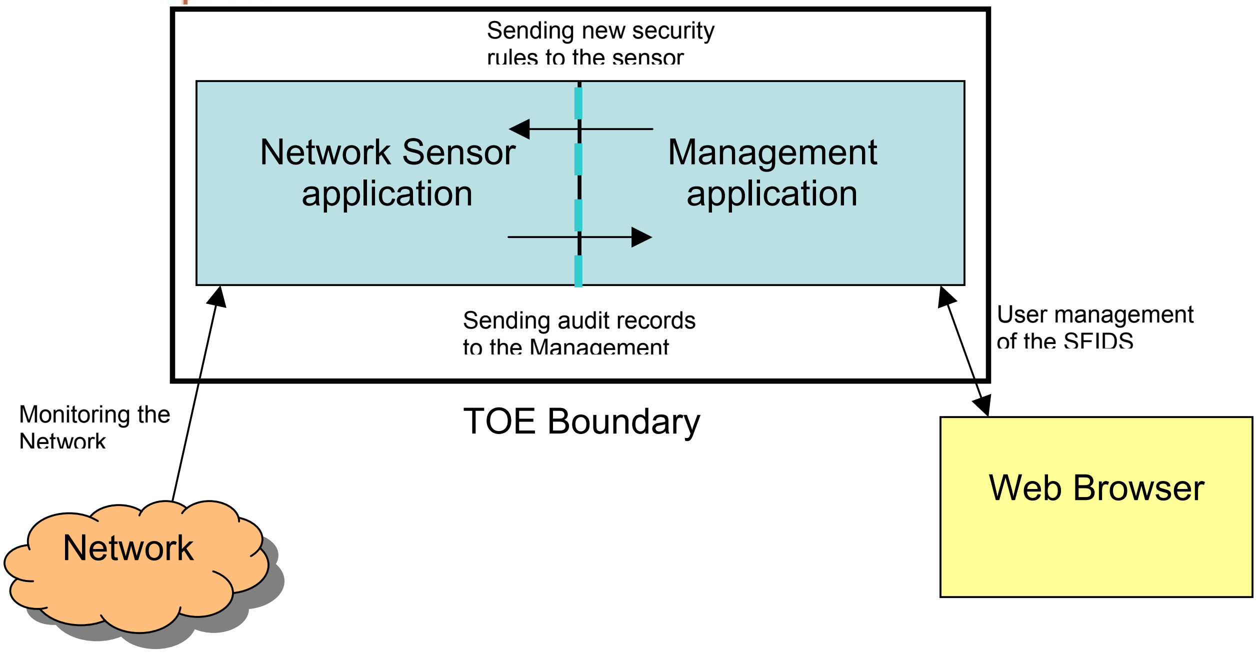
Sending new security rules to the sensor
Network Sensor application
Management application
Sending audit records to the Management
User management of the SFIDS
Monitoring the Network
TOE Boundary
Web Browser
Network

Figure 2: SFIDS without Sourcefire Management Console

As indicated in the figures above, the system can consist of a single Intrusion Detection Sensor appliance or any number of Intrusion Detection Sensor appliances combined with a single Sourcefire Management Console appliance.

2.4.2 Logical Boundaries

The logical boundaries of the TOE are divided into two groups, one related to the administration and security of the system (Security Audit, Identification and Authentication, Security Management, and Protection of Security Functions), and the other related to the collection and analysis of the network traffic (System Data Collection, Data Analysis and Data Review, Availability and Loss).

2.4.2.1 Security Audit

SFIDS is able to audit the use of administration/management functions of the IDS. This audit is separate from the IDS functionality (recording network traffic), and relates specifically to the management functions of the TOE. This function records attempts to access the system itself, such as successful and failed authentication, as well as the actions taken by the user once authenticated. Auditable actions include changes to the IDS rules and viewing/modifying the audit records of both the system access and the IDS event log.

2.4.2.2 Identification and Authentication

SFIDS requires users to provide unique identification and authentication data (passwords) before any access to the system is granted. The TOE provides five levels of authority for users: Administrator, Rules, Data, Maintenance, and Restrictive Data. An Administrator has complete control over the TOE; they can manage user accounts, create/modify and implement IDS rules, and view/delete the audit records. A user with Rules authority can create or modify IDS rules and they have the ability to implement the rules on the system, but they can not view or delete the audit records. Data users can view/manage/delete the IDS event logs. Restrictive Data users also have these abilities, but only for the network that they have the privilege to manage.

2.4.2.3 Security Management

The SFIDS provides a web-based (using https) management interface for all administration, including the IDS rule set, user accounts and roles, and audit functions.

2.4.2.4 Protection of Security Functions

SFIDS protects the security functions it provides through a variety of mechanisms. One of the primary protections is that users must authenticate before any administrative operations can be performed on the system, including creating new rules or viewing the IDS data.

The IDS collection portion of the SFIDS is protected on the monitored network by "hiding" the fact it is there. This is done primarily by using a non-TCP/IP network stack on the SFIDS, which prevents it from being accessed as a network device on the network. Also, the rule set is protected doubly as the system is configured to not accept any management requests or input from the monitored network.

The TOE protects the ability to continue recording data by periodically clearing the stored logs, starting with the oldest records first. This assures there is always adequate disk space to record current and new data that has been found to match the current rule set.

2.4.2.5 System Data Collection

SFIDS has the ability to set rules to govern the collection of data regarding potential intrusions. While SFIDS contains default rules to detect currently known vulnerabilities and exploits, new rules can be created to detect new vulnerabilities as well as specific network traffic, allowing the administrator complete control over the types of traffic that will be monitored.

2.4.2.6 System Data Analysis

To analyze the data collected by snort, SFIDS uses signatures and preprocessors. Signatures are patterns of traffic that can be used to detect potential attacks or exploits. Since many attacks or exploits require several network connections to work, the IDS also provides the ability to detect these more complex patterns through preprocessors that are included in the TOE. The TOE embodies signatures and preprocessors in rules that can be designed and exercised by the TOE.

The administrator can manage the signature identification capabilities by adding and editing rules to respond to the latest exploits. Also, based upon results of analysis, the administrator can trigger alarms for notification of a problem.

2.4.2.7 System Data Review, Availability and Loss

IDS event logs can only be viewed by authorized users (Administrator and Data roles). The data stores of the raw collection data are constantly monitored and if they become too full, new records will replace the oldest records to prevent active/current data loss.

3. Security Environment

The IDS System PP provides the following policies, threats and assumptions about the TOE.

3.1 Threats to Security

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.

3.1.1 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.

3.1.2 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.

T.EXPOSE An improperly configured IT environment may allow unauthorized users to gain access to the TSF.

3.2 Organization Security Policies

The following policies apply to the TOE and the intended environment of the TOE.

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.

3.3 Secure Usage Assumptions

The following usage assumptions are made about the intended environment of the TOE.

3.3.1 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.

3.3.2 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.

3.3.3 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. Security Objectives

This section defines the security objectives of the TOE and its supporting environment. These security objectives, categorized as either IT security objectives for the TOE or its environment are taken from the IDS System PP. All of the identified organizational policies are addressed by the security objectives described below.

4.1 IT Security Objectives for the TOE

The following security objectives are intended to be satisfied by the TOE.

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.EXPORT When any IDS component makes its data available to another IDS components, the TOE will ensure the confidentiality of the System data.

4.2 Security Objectives for the Environment

The following security objectives for the environment of the TOE must be satisfied in order for the TOE to fulfill its own security objectives. These objectives do not levy any IT requirements but are satisfied by procedural or administrative measures.

O.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.

O.PHYCAL Those responsible for the TOE must ensure that those parts of the TOE critical to security policy are protected from any physical attack.

O.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.

O.PERSON Personnel working as authorized administrators shall be carefully selected and trained for proper operation of the System.

O.INTROP The TOE is interoperable with the IT System it monitors.

5. IT Security Requirements

This section provides a list of all security functional requirements for the TOE as taken from the IDSSPP. This PP does not identify any requirements for the IT environment.

5.1 TOE Security Functional Requirements

The following table lists the SFRs required to satisfy the IDS System PP.

Security Functional Class	Security Functional Components
Security audit (FAU)	Audit data generation (FAU_GEN.1)
	Audit review (FAU_SAR.1)
	Restricted audit review (FAU_SAR.2)
	Selectable audit review (FAU_SAR.3)
	Selective audit (FAU_SEL.1)
	Guarantees of audit data availability (FAU_STG.2)
	Prevention of audit data loss (FAU_STG.4)
Identification and authentication (FIA)	Timing of authentication (FIA_UAU.1)
	User attribute definition (FIA_ATD.1)
	Timing of authentication (FIA_UID.1)
Security management (FMT)	Management of security functions behavior (FMT_MOF.1)
	Management of TSF data (FMT_MTD.1)
	Security roles (FMT_SMR.1)
Protection of the TSF (FPT)	Basic internal TSF data transfer protection (FPT_ITT.1)
	Non-bypassability of the TSP (FPT_RVM.1)
	TSF domain separation (FPT_SEP.1)
	Reliable time stamps (FPT_STM.1)
Intrusion Detection System (IDS)	System Data Collection (IDS_SDC.1)
	Analyzer analysis (IDS_ANL.1)
	Analyzer react (IDS_RCT.1)
	Restricted Data Review (IDS_RDR.1)
	Guarantee of System Data Availability (IDS_STG.1)
	Prevention of System data loss (IDS_STG.2)

Table 1 Security Functional Components

5.1.1 Security audit (FAU)

5.1.1.1 Audit data generation (FAU_GEN.1)

5.1.1.1.1 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 (see Table 2); and

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

Component	Event	Details
FAU_GEN.1	Start-up and shutdown of audit functions	
FAU_GEN.1	Access to System	
FAU_GEN.1	Access to the TOE and System data	Object IDS, Requested access
FAU_SAR.1	Reading of information from the audit records	
FAU_SAR.2	Unsuccessful attempts to read information from the audit records	
FAU_SEL.1	All modifications to the audit configuration that occur while the audit collection functions are operating	
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	
FMT_MTD.1	All modifications to the values of TSF data	
FMT_SMR.1	Modifications to the group of users that are part of a role	User identity

Table 2 Auditable Events

Note: The IDS_SDC and IDS_ANL requirements address the recording of results from IDS scanning, sensing, and analyzing tasks (i.e., System data).

5.1.1.1.2 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

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 Table 2 Auditable Events.

5.1.1.2 Audit review (FAU_SAR.1)

5.1.1.2.1 FAU_SAR.1.1

The TSF shall provide [**users with the Administrator or Maintenance Role**] with the capability to read [**all audit information**] from the audit records.

5.1.1.2.2 FAU_SAR.1.2

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

5.1.1.3 Restricted audit review (FAU_SAR.2)

5.1.1.3.1 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.

5.1.1.4 Selectable audit review (FAU_SAR.3)

5.1.1.4.1 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.

5.1.1.5 Selective audit (FAU_SEL.1)

5.1.1.5.1 FAU_SEL.1.1

The TSF shall be able to include or exclude auditable events from the set of audited events based on the following attributes:

a) event type;
b) [**no other attributes**].

5.1.1.6 Guarantees of audit data availability (FAU_STG.2)

5.1.1.6.1 FAU_STG.2.1

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

5.1.1.6.2 FAU_STG.2.2

The TSF shall be able to detect modifications to the audit records **in the audit trail**.[1]

5.1.1.6.3 FAU_STG.2.3

The TSF shall ensure that [**the most recent, limited by available audit storage, at least one**] audit records will be maintained when the following conditions occur: [***audit storage exhaustion***].

5.1.1.7 Prevention of audit data loss (FAU_STG.4)

5.1.1.7.1 FAU_STG.4.1

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

5.1.2 Identification and authentication (FIA)

5.1.2.1 Timing of authentication (FIA_UAU.1)

5.1.2.1.1 FIA_UAU.1.1

The TSF shall allow [**entry of identification and authentication data**] on behalf of the user to be performed before the user is authenticated.

[1] This requirement has been modified to conform with International Interpretations #141 and #202.

5.1.2.1.2 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.

5.1.2.2 User attribute definition (FIA_ATD.1)

5.1.2.2.1 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) Authorizations; and
d) [**None**].

5.1.2.3 Timing of identification (FIA_UID.1)

5.1.2.3.1 FIA_UID.1.1

The TSF shall allow [**entry of identification and authentication data**] on behalf of the user to be performed before the user is identified.

5.1.2.3.2 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.

5.1.3 Security management (FMT)

5.1.3.1 Management of security functions behavior (FMT_MOF.1)

5.1.3.1.1 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.

5.1.3.2 Management of TSF data (FMT_MTD.1)

5.1.3.2.1 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 [

Data role	**- query and delete IDS event data;**
Rules role	**- add, edit, and implement rules;**
Administrator role	**- all functions (query audit data, create and implement rules and manage users);**
Maintenance role	**- view status, manage audit logs, and manage system maintenance utilities;**
Restrictive Data role	**- query and delete IDS event logs (limited to only data that they are privileged to access)**].

5.1.3.3 Security roles (FMT_SMR.1)

5.1.3.3.1 FMT_SMR.1.1

The TSF shall maintain the following *roles:* ~~authorized~~ Administrator, authorized System administrators, and [**Rules, maintenance, restricted data, and Data**].

5.1.3.3.2 FMT_SMR.1.2

The TSF shall be able to associate users with roles.

5.1.4 Protection of the TOE security functions (FPT)

5.1.4.1 Non-bypassability of the TSP (FPT_RVM.1)

5.1.4.1.1 FPT_RVM.1.1

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

5.1.4.2 TSF domain separation (FPT_SEP.1)

5.1.4.2.1 FPT_SEP.1.1

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

5.1.4.2.2 FPT_SEP.1.2

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

5.1.4.3 Reliable time stamps (FPT_STM.1)

5.1.4.3.1 FPT_STM.1.1

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

5.1.4.4 Basic internal TSF data transfer protection (FPT_ITT.1)

5.1.4.4.1 FPT_ITT.1.1

The TSF shall protect TSF data from [***disclosure, modification***] when it is transmitted between separate parts of the TOE.

5.1.5 IDS Component Requirements (IDS)

5.1.5.1 System Data Collection (EXP) (IDS_SDC.1)

5.1.5.1.1 IDS_SDC.1.1

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

a) [***network traffic, and detected known vulnerabilities***]; and
b) [**no other events**]. **(EXP)**

5.1.5.1.2 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 Table 3 System Events. **(EXP)**

Component	Event	Details
IDS_SDC.1	Start-up and shutdown	None

Component	Event	Details
IDS_SDC.1	Identification and authentication events	User identity, location, source address, destination address
IDS_SDC.1	Data accesses	Object IDS, requested access, source address, destination address
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	Start-up and shutdown of audit functions	None
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	Authentication configuration	Account names for cracked passwords, account policy parameters
IDS_SDC.1	Accountability policy configuration	Accountability policy configuration parameters
IDS_SDC.1	Detected known vulnerabilities	Identification of the known vulnerability

Table 3 System Events

5.1.5.2 Analyzer analysis (EXP) (IDS_ANL.1)

5.1.5.2.1 IDS_ANL.1.1

The System shall perform the following analysis function(s) on all IDS data received:

a) [***signature***]; and
b) [**port scan**
c) **HTTP Decode**
d) **Packet Defragmentation**
e) **Stateful Inspection**
f) **Telnet Decode**]. **(EXP)**

5.1.5.2.2 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) [**The packets analyzed to determine the result**]. **(EXP)**

5.1.5.3 Analyzer react (EXP) (IDS_RCT.1)

5.1.5.3.1 IDS_RCT.1.1

The System shall send an alarm to [**a defined email administrative address**] and take [**no further action**] when an intrusion is detected. **(EXP)**

5.1.5.4 Restricted Data Review (EXP) (IDS_RDR.1)

5.1.5.4.1 IDS_RDR.1.1

The System shall provide [**users with Administrator or Data roles**] with the capability to read [**all captured IDS data**] from the System data. **(EXP)**

5.1.5.4.2 IDS_RDR.1.2

The System shall provide the System data in a manner suitable for the user to interpret the information. **(EXP)**

5.1.5.4.3 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. **(EXP)**

5.1.5.5 Guarantee of System Data Availability (EXP) (IDS_STG.1)

5.1.5.5.1 IDS_STG.1.1

The System shall protect the stored System data from unauthorized deletion. **(EXP)**

5.1.5.5.2 IDS_ STG.1.2

The System shall protect the stored System data from modification. **(EXP)**

5.1.5.5.3 IDS_ STG.1.3

The System shall ensure that [**the most recent, limited by available System data storage, at least one**] System data will be maintained when the following conditions occur: [***System data storage exhaustion***]. **(EXP)**

5.1.5.6 Prevention of System data loss (EXP) (IDS_STG.2)

5.1.5.6.1 IDS_STG.2.1

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

5.2 TOE Security Assurance Requirements

The security assurance requirements for the TOE are the Evaluation Assurance Level (EAL) 2 components as specified in Part 3 of the Common Criteria. No operations are applied to the assurance components.

Assurance Class	Assurance Components
Configuration Management (ACM)	ACM_CAP.2 Configuration items
Delivery and Operation (ADO)	ADO_DEL.1 Delivery procedures
	ADO_IGS.1 Installation, generation, and start-up procedures
Development (ADV)	ADV_FSP.1 Informal Function Specification
	ADV_HLD.1 Descriptive high-level design
	ADV_RCR.1 Informal correspondence demonstration
Guidance Documents (AGD)	AGD_ADM.1 Administrator guidance
	AGD_USR.1 User guidance
Tests (ATE)	ATE_COV.1 Evidence of coverage
	ATE_FUN.1 Functional testing
	ATE_IND.2 Independent testing - sample
Vulnerability assessment (AVA)	AVA_SOF.1 Strength of TOE security function evaluation
	AVA_VLA.1 Developer vulnerability analysis

Table 4 EAL2 Assurance Components

5.2.1 Configuration Management (ACM)

5.2.1.1 Configuration Items (ACM_CAP.2)

5.2.1.1.1 ACM_CAP.2.1D

The developer shall provide a reference for the TOE.

5.2.1.1.2 ACM_CAP.2.2D

The developer shall use a CM system.

5.2.1.1.3 ACM_CAP.2.3D

The developer shall provide CM documentation.

5.2.1.1.4 ACM_CAP.2.1C

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

5.2.1.1.5 ACM_CAP.2.2C

The TOE shall be labeled with its reference.

5.2.1.1.6 ACM_CAP.2.3C

The CM documentation shall include a configuration list.

5.2.1.1.7 ACM_CAP.2.4C

The configuration list shall describe the configuration items that comprise the TOE.

5.2.1.1.8 ACM_CAP.2.5C

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

5.2.1.1.9 ACM_CAP.2.6C

The CM system shall uniquely identify all configuration items.

5.2.1.1.10 ACM_CAP.2.7C

The configuration list shall uniquely identify all configuration items that comprise the TOE.[2]

5.2.1.1.11 ACM_CAP.2.1E

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

5.2.2 Delivery and Operation (ADO)

5.2.2.1 Delivery Procedures (ADO_DEL.1)

5.2.2.1.1 ADO_DEL.1.1D

The developer shall document procedures for delivery of the TOE or parts of it to the user.

[2] This requirement has been modified to comply with International Interpretation #3.

5.2.2.1.2 ADO_DEL.1.2D

The developer shall use the delivery procedures.

5.2.2.1.3 ADO_DEL.1.1C

The delivery documentation shall describe all procedures that are necessary to maintain security when distributing versions of the TOE to a user's site.

5.2.2.1.4 ADO_DEL.1.1E

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

5.2.2.2 Installation, generation, and start-up procedures (ADO_IGS.1)

5.2.2.2.1 ADO_IGS.1.1D

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

5.2.2.2.2 ADO_IGS.1.1C

The **installation, generation and start-up** documentation shall describe **all** the steps necessary for secure installation, generation, and start-up of the TOE.[3]

5.2.2.2.3 ADO_IGS.1.1E

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

5.2.2.2.4 ADO_IGS.1.2E

The evaluator shall determine that the installation, generation, and start-up procedures result in a secure configuration.

5.2.3 Development (ADV)

5.2.3.1 Informal Function Specification (ADV_FSP.1)

5.2.3.1.1 ADV_FSP.1.1D

The developer shall provide a functional specification.

5.2.3.1.2 ADV_FSP.1.1C

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

5.2.3.1.3 ADV_FSP.1.2C

The functional specification shall be internally consistent.

[3] This requirement has been modified to comply with International Interpretation #51.

5.2.3.1.4 ADV_FSP.1.3C

The functional specification shall describe the purpose and method of use of all external TSF interfaces, providing details of effects, exceptions and error messages, as appropriate.

5.2.3.1.5 ADV_FSP.1.4C

The functional specification shall completely represent the TSF.

5.2.3.1.6 ADV_FSP.1.1E

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

5.2.3.1.7 ADV_FSP.1.2E

The evaluator shall determine that the functional specification is an accurate and complete instantiation of the TOE security requirements.

5.2.3.2 Descriptive high-level design (ADV_HLD.1)

5.2.3.2.1 ADV_HLD.1.1D

The developer shall provide the high level design of the TSF.

5.2.3.2.2 ADV_HLD.1.1C

The presentation of the high level design shall be informal.

5.2.3.2.3 ADV_HLD.1.2C

The high level design shall be internally consistent.

5.2.3.2.4 ADV_HLD.1.3C

The high level design shall describe the structure of the TSF in terms of subsystems.

5.2.3.2.5 ADV_HLD.1.4C

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

5.2.3.2.6 ADV_HLD.1.5C

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

5.2.3.2.7 ADV_HLD.1.6C

The high level design shall identify all interfaces to the subsystems of the TSF.

5.2.3.2.8 ADV_HLD.1.7C

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

5.2.3.2.9 ADV_HLD.1.1E

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

5.2.3.2.10 ADV_HLD.1.2E

The evaluator shall determine that the high level design is an accurate and complete instantiation of the TOE security requirements.

5.2.3.3 Informal correspondence demonstration (ADV_RCR.1)

5.2.3.3.1 ADV_RCR.1.1D

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

5.2.3.3.2 ADV_RCR.1.1C

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

5.2.3.3.3 ADV_RCR.1.1E

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

5.2.4 Guidance Documents (AGD)

5.2.4.1 Administrator Guidance (AGD_ADM.1)

5.2.4.1.1 AGD_ADM.1.1D

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

5.2.4.1.2 AGD_ADM.1.1C

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

5.2.4.1.3 AGD_ADM.1.2C

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

5.2.4.1.4 AGD_ADM.1.3C

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

5.2.4.1.5 AGD_ADM.1.4C

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

5.2.4.1.6 AGD_ADM.1.5C

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

5.2.4.1.7 AGD_ADM.1.6C

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

5.2.4.1.8 AGD_ADM.1.7C

The administrator guidance shall be consistent with all other documents supplied for evaluation.

5.2.4.1.9 AGD_ADM.1.8C

The administrator guidance shall describe all security requirements on the IT environment that are relevant to the administrator.

5.2.4.1.10 AGD_ADM.1.1E

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

5.2.4.2 User Guidance (AGD_USR.1)

5.2.4.2.1 AGD_USR.1.1D

The developer shall provide user guidance.

5.2.4.2.2 AGD_USR.1.1C

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

5.2.4.2.3 AGD_USR.1.2C

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

5.2.4.2.4 AGD_USR.1.3C

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

5.2.4.2.5 AGD_USR.1.4C

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

5.2.4.2.6 AGD_USR.1.5C

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

5.2.4.2.7 AGD_USR.1.6C

The user guidance shall describe all security requirements on the IT environment that are relevant to the user.

5.2.4.2.8 AGD_USR.1.1E

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

5.2.5 Security Testing (ATE)

5.2.5.1 Evidence of coverage (ATE_COV.1)

5.2.5.1.1 ATE_COV.1.1D

The developer shall provide evidence of the test coverage.

5.2.5.1.2 ATE_COV.1.1C

The evidence of the test coverage shall show the correspondence between the tests identified in the test documentation and the TSF as described in the functional specification.

5.2.5.1.3 ATE_COV.1.1E

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

5.2.5.2 Functional testing (ATE_FUN.1)

5.2.5.2.1 ATE_FUN.1.1D

The developer shall test the TSF and document the results.

5.2.5.2.2 ATE_FUN.1.2D

The developer shall provide test documentation.

5.2.5.2.3 ATE_FUN.1.1C

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

5.2.5.2.4 ATE_FUN.1.2C

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

5.2.5.2.5 ATE_FUN.1.3C

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

5.2.5.2.6 ATE_FUN.1.4C

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

5.2.5.2.7 ATE_FUN.1.5C

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

5.2.5.2.8 ATE_FUN.1.1E

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

5.2.5.3 Independent testing – sample (ATE_IND.2)

5.2.5.3.1 ATE_IND.2.1D

The developer shall provide the TOE for testing.

5.2.5.3.2 ATE_IND.2.1C

The TOE shall be suitable for testing.

5.2.5.3.3 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.

5.2.5.3.4 ATE_IND.2.1E

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

5.2.5.3.5 ATE_IND.2.2E

The evaluator shall test a subset of the TSF as appropriate to confirm that the TOE operates as specified.

5.2.5.3.6 ATE_IND.2.3E

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

5.2.5.4 Strength of TOE security function evaluation (AVA_SOF.1)

5.2.5.4.1 AVA_SOF.1.1D

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

5.2.5.4.2 AVA_SOF.1.1C

For each mechanism with a strength of TOE security function claim the strength of TOE security function analysis shall show that it meets or exceeds the minimum strength level of SOF-Basic.

5.2.5.4.3 AVA_SOF.1.2C

For each mechanism with a specific strength of TOE security function claim the strength of TOE security function analysis shall show that it meets or exceeds the specific strength of function metric SOF-Basic.

5.2.5.4.4 AVA_SOF.1.1E

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

5.2.5.4.5 AVA_SOF.1.2E

The evaluator shall confirm that the strength claims are correct.

5.2.5.5 Developer vulnerability analysis (AVA_VLA.1)

5.2.5.5.1 AVA_VLA.1.1D

~~The developer shall perform and document an analysis of the TOE deliverables searching for obvious ways in which a user can violate the TSP.~~ **The developer shall perform a vulnerability analysis.**[4]

5.2.5.5.2 AVA_VLA.1.2D

~~The developer shall document the disposition of obvious vulnerabilities.~~ **The developer shall provide vulnerability analysis documentation.** [5]

5.2.5.5.3 AVA_VLA.1.1C

~~The documentation shall show, for all identified vulnerabilities, that the vulnerability cannot be exploited in the intended environment for the TOE.~~ **The vulnerability analysis documentation shall describe the analysis of the TOE deliverables performed to search for obvious ways in which a user can violate the TSP.**[6]

5.2.5.5.4 AVA_VLA.1.2C

The vulnerability analysis documentation shall describe the disposition of obvious vulnerabilities.[7]

5.2.5.5.5 AVA_VLA.1.3C

The vulnerability analysis documentation shall show, for all identified vulnerabilities, that the vulnerability cannot be exploited in the intended environment for the TOE. [8]

5.2.5.5.6 AVA_VLA.1.1E

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

5.2.5.5.7 AVA_VLA.1.2E

The evaluator shall conduct penetration testing, building on the developer vulnerability analysis, to ensure obvious vulnerabilities have been addressed.

[4] This requirement has been modified to comply with International Interpretation #51.
[5] This requirement has been modified to comply with International Interpretation #51.
[6] This requirement has been modified to comply with International Interpretation #51.
[7] This requirement has been added to comply with International Interpretation #51.
[8] This requirement has been added to comply with International Interpretation #51.

6. TOE Summary Specification

This chapter describes the security functions and associated assurance measures.

6.1 TOE Security Functions

Each of the security function descriptions is organized by the security requirements corresponding to the security function. Hence, each function is described by describing how it specifically satisfies each of its related requirements. This serves to both describe the security functions and rationalize that the security functions are suitable to satisfy the necessary requirements.

6.1.1 Security Audit

FAU_GEN.1 Audit Data Generation

Auditing is the recording of events within the system. SFIDS records two classes of events: security events and IDS events. IDS events are dealt with separately under the System Data security functions. Security events relate to the proper functioning and use of the system, and allow an administrator to track the management functions performed.

The following events are stored in the security log:

a) Startup and shutdown of the audit function

b) Access to the System

c) Access to the TOE and System data

d) Viewing of the audit records

e) Unsuccessful attempts to view the audit records

f) All modification to the audit configuration that occur during collection

g) All identification and authentication attempts, including the user and location where authentication was attempted

h) All modification to the behavior of the TSF

i) All modifications to TSF data values

j) Modification of user accounts, creation, deletion, and modifications

Each audit record contains the following information: date and time the event occurred, the type of event(subsystem), identity of the user, and the result of the event (message). The message identifies the action that the user attempted when logged on. Only the message field in the audit records for logging in and out of the TOE details the success or failure of the attempt. After successfully logging on, additional audit logs are used to indicate actions taken during the usage of the TOE.

FAU_SAR.1 Audit Review

SFIDS provides the ability for user with the Administrator or Maintenance role to view security audit data for the system. The audit logs are viewable through the standard management interface.

FAU_SAR.2 Restricted Audit Review

No security related actions can be taken without a successful user authentication and the management interface allows only users who have the Administrator role to view the audit records.

FAU_SAR.3 Selectable Audit Review

While viewing the security audit records, the audit review interface, available from the Management Console, provides the ability to sort the data for display based upon the following properties:

- Date and time
- User
- Type of event (Subsystem)
- Success or Failure of the event for the login only, all other records identify what action occurred (Message)

FAU_SEL.1 Selectable Audit

SFIDS provides root access to the command line so suppression lists can be created for audit events based on IP address, message, subsystem, and username. Any one of these four types can be suppressed, preventing the audit event from being generated.

FAU_STG.2 Guarantees of Data Availability

The only way to access the audit records is through the management console. The TOE provides protection for the security audit records primarily by preventing access to the system without successful authentication. Subsequently, the TOE requires that a user must have the Administrator or Maintenance role before granting access to the audit records via audit record management function interfaces. Further, since the audit function starts automatically with the TOE, and cannot be disabled, all selected (see FAU_SEL.1) actions are recorded, including possible modification to the records.

As indicated below, when available audit storage is exhausted the TOE automatically overwrites the oldest audit events. This ensures that the most recent audit events limited only by the size of the audit trail, but at least one audit event, is always available.

FAU_STG.4 Prevention of Audit Data Loss

When the TOE begins to run out of storage space for the audit records (85% disk capacity allotted for record storage) or the security event database limit of 100,000 has been reached, a warning is sent to the designated administrator via email to inform them of the loss. If the audit process runs out of disk space or the limit is exceeded, then the oldest current log files will be automatically overwritten to prevent new actions from occurring without being tracked. This can occur if the event database, security databases, or the log files grow and exceed the 85% limit.

6.1.2 Identification and Authentication

FIA_ATD.1 User Attribute Definition

User account information is stored in the TOE with the following attributes: user name, authentication data (password), and their assigned role(s) (authorizations). User accounts can have multiple roles assigned to them that allows them a broader and more focused set of privileges. The user account information is stored in a TSF database that is modifiable only by an administrator.

FIA_UAU.1 Timing of Authentication and **FIA_UID.1 Timing of Authentication**

SFIDS requires users to provide unique identification and authentication data (passwords) before any access to the system is granted. When identification and authentication data is entered, the TOE attempts to identify the applicable user account from the provided identity and if a match is found the password provided is compared against that stored with the user account information. If a user account cannot be associated with the provided identity or the provided password does not match that stored with the user account information, identification and authentication will fail. No actions are allowed, other than entry of identification and authentication data, until successful identification and authentication.

6.1.3 Security Management

FMT_MOF.1 Management of Security Functions Behavior

SFIDS requires user authentication before any actions can be performed (other than entry of identification and authentication data) on the TOE, security-related or otherwise. Due to this, only authenticated users can access any functions on the system. Users with the "Rules" or "Administrator" role have the ability to create, modify, and implement rules that collect IDS events. Note that the Administrator role is a specific instance of the more general "authorized System administrators" role per FMT_SMR.1.

FMT_MTD.1 Management of TSF Data

See FMT_SMR.1.

FMT_SMR.1 Security Roles

The TOE has five defined roles, each with its own set of privileges. When a new user account is created, it must be assigned a role.

- "Administrator" Role: this role can perform all management functions on the TOE. A user with this role can manage user accounts (create, delete, modify), view, query and delete the security and IDS event logs, and manage the rules that govern the IDS.
- "Rules" role: this role can create, modify, and implement existing rules for the IDS. This role cannot perform other functions.
- "Restricted Data" role: this role can view and manage IDS events that they have access to through predefined restrictions. This restriction defines what events they are privileged to view.
- "Maintenance" role: this role can view and manage status, security audit events, system time, and the reporting functionality of the product. Additionally, they can perform system level maintenance related actions.
- "Data" role: this role can view and manage IDS events including deleting old IDS events as necessary.

Note that the "authorized System administrators" role defined FMT_SMR.1 (from the IDSSPP) is a generalization of the Administrator role defined in the TOE. Hence, a user that assumes the Administrator role is serving in the "Administrator" and an "authorized System administrator" capacity simultaneously.

6.1.4 Protection of Security Functions

FPT_RVM.1 Non-bypassability of the TSP

The TSF requires that all users successfully authenticate before any TSF functions (other than entering identification and authentication data) can be performed. Once a user is identified and authenticated, they are associated with a role that determines which function interfaces the TOE will offer to the user. Each interface is defined to offer specific capabilities, all controlled by the TSF. The TSS does not offer general programming capabilities that might offer the opportunity to attempt to bypass the TSP.

Additionally, the TSF does not accept any commands from or offer any functions to the networks that are monitored by the TOE. This ensures that network entities cannot cause the TOE to not apply its TSPs to applicable network traffic.

FPT_SEP.1 TSF Domain Separation

The TOE software is a pair of applications that operate exclusively on the portion of the TOE hardware providing execution support. Each TOE application operates as a single process that communicates only via network interfaces either to monitor traffic, communicate with another portion of the TOE or to communicate with an administrator. Furthermore, the TOE offers only well defined services at its network interfaces that are specifically designed to only provide the services that are necessary to enforce the TSP and not to offer additional services that might be used to interfere with the operation of the TOE.

FPT_STM.1 Reliable Time Stamps

The TOE uses the system time to generate reliable timestamps for security audit events and is used by snort to generate the timestamp for IDS events The TOE can also receive its time from the Management Console in a distributed environment, which receives its reliable timestamp from its own system time.

FPT_ITT.1 Basic internal TSF data transfer protection

The TSF ensures that data transmitted between separate parts of the TOE are protected from disclosure or modification. This protection is ensured through strong encryption during both setup and the transition of data.. There are four types of communications that can occur between sperate parts of the TOE; event transmission, status updates, remote copy, and remote execution. Event data and status updates both use 256bit SSL encryption to transfer data between two separate TOE devices. Both use Diffie-Hellman with RSA key exchange. The SSL Block encryption uses 256bit AES encryption (DHE-RSA-AES256-SHA). Remote copy (scp) and remote execution (ssh) uses 128 bit AES for data encryption. These transfers are username and password protected to ensure unauthorized access to the TOE.

6.1.5 System Data Collection

IDS_SDC.1 System Data Collection

SFIDS has the ability to set rules to govern the collection of data regarding potential intrusions. While SFIDS contains default rules to detect currently known vulnerabilities and exploits, new rules can be created to detect new vulnerabilities as well as specific network traffic, allowing the administrator complete control over the types of traffic that will be monitored.

As a minimum the SFIDS can collect the following information:

- Network traffic, including protocol, source address, destination address
- Detected known vulnerabilities, including identification of the vulnerability

For each of these data types the following additional information is collected:

- Date and time
- Type of data
- Subject identity (e.g., source address or addresses)
- Outcome of the event[9]

Note that while the IDS_SDC.1 requirement indicates additional information content, that content is dependent upon the data that is collected. Since the TOE only claims to collect network traffic and detected known vulnerabilities, the other information is not relevant.

6.1.6 System Data Analysis

IDS_ANL.1 Analyzer Analysis

To analyze the data collected by the snort, SFIDS uses signatures and preprocessors. Signatures are patterns of traffic that can be used to detect attacks or exploits. Since many attacks or exploits require several network connections to work, the IDS also provides the ability to detect these more complex patterns through preprocessors that are included in the TOE. Note that rules are used to embodied signatures and preprocessors in the TOE.

SFIDS comes with default signatures for known exploits, and the administrator can add new signatures at any time. New signatures are available from the support organization, can also be downloaded from public snort forums, and can be created by the administrator manually. This gives the administrator total control over the detection of traffic, allowing complete customization for the intended environment.

Signatures are used for stateless detections; those intrusion attempts that can be detected with individual packets. Signatures cannot be used to detect intrusions that require multiple packets, such as a Denial of Service attack. To detect these types of events, the IDS uses various preprocessors for stateful inspections, which allow these multi-packet intrusions to be detected. Preprocessors can also provide detection of malformed packets.

[9] It is not clear how an outcome applies to suspected intrusion scenarios. Regardless, it is assumed that any outcome would be implied in the other data collected associated with the potential intrusion.

All signatures entered into the TOE by any means, must conform to this format:

```
<type field> <protocol field> <source IP> <source port> operator <destination IP>
                    <destination port> (option1; option2;)
```

Header – defines the network addresses involved for the traffic to be considered for evaluation by the signature options

- Type - Identifies the action the system should take when a packet triggers the rule.
  - Alert - Sends an alert then logs the details about the packet that triggered the event
  - Pass - Ignores the packet that triggered the event
- Protocol - Specifies the protocol of the packets against which the rule executes.
  - TCP - Executes against traffic using the Transmission Control Protocol
  - UDP - Executes against traffic using the User Datagram Protocol
  - ICMP - Executes against traffic using the Internet Control Message Protocol
  - IP - Executes against traffic using the Internet Protocol
- Source IP - Specifies the source IP address or range of addresses.
  - Any - Executes against packets from any source IP.
  - Numeric IP address - Executes against packets with the specified source IP.
  - CIDR blocks - Executes against packets whose source IP address falls within the specified CIDR block.
- Source Port – Specifies the source port.
  - Any - Executes against traffic with any source port
  - Numeric - Executes against traffic with the specified source port
  - Numeric: numeric - Executes against traffic with the specified range of source ports
  - ! numeric - Executes against traffic with any source port except the port specified after the exclamation point (!)
- Operator - Specifies the direction of the traffic to which the rule applies
  - -> Evaluates all traffic from the source IP to the destination IP
  - <> Evaluates traffic between the source IP to the destination IP
- Destination IP - Specifies the destination IP address or range of addresses
  - Any - Executes against packets with any destination IP. For example, in the following rule, the any in bold specifies any destination IP address: *alert tcp any any -> any any (rest of rule)*
  - Numeric IP address - Executes against packets with the specified destination IP. For example, in the following rule, the numbers in bold specify a specific destination IP address: *alert tcp any any -> 192.168.17.1 any (rest of rule)*
  - CIDR blocks - Executes against packets whose destination IP address falls within the specified CIDR block. For example, in the following rule, the bracketed numbers specify a range of destination IP address: *alert tcp any any -> [10.1.0.0/16,192.168.1.2/24] any (rest of rule)*
- Destination Port - Specifies the destination port
  - Any - Executes against traffic with any destination port
  - Numeric - Executes against traffic with the specified destination port

  - Numeric: numeric - Executes against traffic with the specified range of destination ports
  - ! numeric - Executes against traffic with any destination port except the port specified after the exclamation point (!)

Options – defines the attributes of a packet that must be inspected to determine whether the packet is a match for a specific signature

Options are defined as a keyword and a value, and are listed as keyword:value within the options field of the signature. Following is a list of keywords and their definitions from which their values are derived.

- Ack - Tests the TCP flag value against the value specified in the argument
- Classtype - Identifies the rule classification
- Content - Tests the packet payload content against the pattern specified in the argument
- content-list - Tests the packet payload content against the set of patterns specified in the argument
- Depth - Sets the maximum search depth for a pattern match; this option modifies the content option
- Distance - Indicates that the next content match must be at least the specified number of bytes from the last content match
- Dsize - Tests the packet's payload size against the value specified in the argument
- Flags - Tests the TCP flags against the value specified in the argument
- Flow - Allows rules to only apply to the direction of the traffic flow specified in the argument (used in conjunction with TCP stream reassembly)
  - to_client - triggers on server responses from A to B
  - to_server - trigger on client requests from A to B
  - from_client - triggers on client requests from A to B
  - from_server - triggers on server responses from A to B
  - established - triggers only on established TCP connections
  - stateless - trigges regardless of the state of the stream processor (useful for packets that are designed to cause machines to crash)
  - no_stream - do not trigger on "rebuilt" stream packets (useful for Dsize and stream4)
  - only_stream - only triggers on "rebuilt" stream packets
- Fragbits - Tests the fragmentation bits of the IP header
- icmp_id - Tests the ICMP echo ID value against the value specified in the argument
- icmp_seq - Tests the ICMP echo sequence number against the value specified in the argument
- icode - Tests the ICMP code value against the value specified in the argument
- ID - Tests the IP header's fragment ID field value against the value specified in the argument
- IPoption - Tests the IP option fields against the codes specified in the argument
- ip_proto - Tests the IP header's protocol value against the value specified in the argument
- itype - Tests the ICMP type value against the value specified in the argument
- Msg - Prints the message specified in the argument in events and packet logs
- Nocase - Matches the preceding content string with case insensitivity

- Offset - Sets the offset value to begin attempting a pattern match; this option modifies the content option
- Priority - Identifies the rule severity
- Rawbytes - Indicates that Snort should ignore the decoded packet and match against the raw payload data (used in rules that check for telnet option negotiation codes)
- Reference - References an external attack ID
- RPC - Tests RPC services against application/procedure calls specified in the argument
- Sameip - Determines if the source IP equals the destination IP
- Seq - Tests the TCP sequence number against the value specified in the argument
- Session - Dumps the application layer information for the session
- Stateless - Specifies that the rule is valid regardless of stream state
- Tag - The associated argument identifies the advanced logging actions for data about traffic that triggers the rule
- TOS - Tests the IP header's TOS field value against the value specified in the rule argument
- TTL - Tests the IP header's TTL field value against the value specified in the rule argument
- uricontent - Searches for a pattern specified in the argument in the URI portion of a packet
- Within - Indicates that the next content match must be within the specified number of bytes from the last content match

The following preprocessors are available for the detection of stateful or malformed intrusions:

Back Orifice Detection

This preprocessor searches for packets that can show the presence of Back Orifice, or attempts to install Back Orifice onto computers on the network.

Checksum Verification

This preprocessor verifies the size of packets being sent to the network, detecting malformed packets that may be used in various attacks.

IP Defragmentation

This preprocessor enables the TOE to rebuild packets that have been fragmented by the network prior to inspection against other preprocessors and signatures.

HTTP Normalization

This preprocessor decodes the URI portion of http packets into non-obfuscated ASCII that can then be used for evaluation against signatures.

RPC Normalization

This preprocessor decodes RPC traffic for analysis against signatures (similar to HTTP Normalization).

Stateful Inspection and Stream Reassembly

This preprocessor provides stateful inspection of packets, allowing detection of intrusion attempts that span multiple packets. The Stream reassembly allows detection of sessions between clients and servers, and then the analysis of this traffic for specific patterns.

Telnet Normalization

This preprocessor decodes Telnet traffic for analysis against signatures (similar to HTTP Normalization).

When a pattern of traffic has been matched to a signature or a preprocessor, the specific event is recorded in the System Data log where it can be viewed by users with either the Administrator or Data roles. The events are logged with the following information: the event type and signature or preprocessor match, the time and date of the event, the data source and a copy of the packets used to identify the pattern.

IDS_RCT.1 Analyzer React

When signature matches are found, they can either be logged for later use of set to trigger an alarm. This is part of the configuration of an active signature. If a signature has been marked to trigger an alarm, an automatic email is sent to a specified email address detailing the type of event and the time it occurred.

6.1.7 System Data Review, Availability and Loss

IDS_RDR.1 Restricted Data Review

In SFIDS, only successfully authenticated users can access the TOE, and then only users with either the Administrator or Data roles can view the IDS events collected and analyzed by snort. The data gathered by snort is transferred to the Management Console and there interpreted into a readable format for the user. The data is then viewed through the normal web-based management interface. Users with the Rules role are unable to access the System Data logs.

IDS_STG.1 Guarantee of System Data Availability

SFIDS protects the gathered system data logs from unauthorized modification or deletion by presenting only the web-based interface to all users. No users are allowed to edit the logs; they are marked for read-only access, preventing user modification. Only users with either the Administrator, Restrictive Data, or Data roles can delete the logs.

To guarantee that sufficient storage space is always available for incoming/new events, there are three mechanisms that achieve this. First, when the disk space reaches 85% of its capacity, the oldest log files will be deleted, keeping at least 15% free disk space. Secondly, a user interface setting allows for a limit for the number of events that will be stored in the database. The default is 1,000,000 events and when this limit is reached, the oldest events will be deleted ensuring this limit is maintained. These three mechanisms maintain the system disk space in order to handle the case when a flood of data comes in before overwriting can occur and always ensures that more than one event can always be added in the log.

IDS_STG.2 Prevention of System Data Loss

To prevent the loss in new/current event data, there are three mechanisms to limit event data loss, event database limit, audit record limit (refer to FAU_STG.4 "Prevention of Audit Data Loss" for more detail), and disk capacity. The event database size limit overwrites the oldest events when it reaches the default of 1,000,000 event records. This limits the number of events that can be stored in the database and allows for new event insertions. When the disk space reaches 85% of its capacity, the TSF will delete as few log files as possible to keep the space below 85% capacity. This can occur when any new data is stored on the disk, regardless of what database it is being written to. When any one of these is set to occur, the users designated as the Administrator role will receive an email about the issues automatically until they intervene.

6.2 TOE Security Assurance Measures

The following assurance measures are applied to satisfy the Common Criteria EAL2 assurance requirements:

- Process Assurance;
- Delivery and Guidance;
- Design Documentation;
- Tests; and
- Vulnerability Assessment.

6.2.1 Process Assurance

6.2.1.1 Configuration Management

The configuration management measures applied by Sourcefire ensure that configuration items are uniquely identified. Sourcefire ensures changes to the implementation representation are controlled and that TOE associated configuration item modifications are properly controlled. Sourcefire performs configuration management on the TOE implementation representation, design, tests, user and administrator guidance, vulnerability assessment, delivery, installation, and the CM documentation. These activities are documented in:

- Sourcefire Intrusion Detection System Configuration Management Plan

The Configuration Management assurance measure satisfies the ACM_CAP.2 assurance requirements

6.2.2 Delivery and Guidance

Sourcefire provides delivery documentation that explains how the TOE is delivered and procedures to identify the TOE, allow detection of unauthorized modifications of the TOE and installation and generation instructions at start-up. Sourcefire's delivery procedures describe the steps to be used for the secure installation, generation, and start-up of the TOE. These procedures are documented in:

- Sourcefire Intrusion Detection System Delivery Procedures

Sourcefire provides guidance on how to properly utilize the TOE security functions, including function descriptions, warnings, effects, assumptions, etc. The installation and generation procedures, included in the administrator guidance, describe the steps necessary to install Sourcefire IDS products in accordance with the evaluated configuration. Note that there are no conventional "users" of Sourcefire products. As such, all applicable guidance for "administrator" and "users" is embodied the following guides:

- Sourcefire Network Sensor 2000 Installation Guide Version 3.2.3
- Sourcefire Network Sensor 2100 Installation Guide Version 3.2.3
- Sourcefire Network Sensor 3000 Installation Guide Version 3.2.3
- Sourcefire Network Sensor 1000 Installation Guide Version 3.2.3
- Sourcefire Network Sensor 500 Installation Guide Version 3.2.3
- Sourcefire Management Console 1000 Installation Guide Version 3.2.3
- Sourcefire Management Console 3000 Installation Guide Version 3.2.3
- Sourcefire Network Sensor User Guide Version 3.2.3
- Sourcefire Management Console User Guide Version 3.2.3

The Delivery and Guidance assurance measure satisfies the following Assurance requirements:

- ADO_DEL.1;
- ADO_IGS.1;
- AGD_ADM.1; and,
- AGD_USR.1.

6.2.3 Design Documentation

The Design Documentation provided for SFIDS is provided in four documents:

- Sourcefire Intrusion Detection System High Level Design: Sensor
- Sourcefire Intrusion Detection System High Level Design: Management Console

- Sourcefire Intrusion Detection System Functional Specification: Management Console
- Sourcefire Intrusion Detection System Functional Specification: Network Sensor

These documents serve to describe the security functions of the TOE, its interfaces both external and between subsystems, the architecture of the TOE (in terms of subsystems), and correspondence between the available design abstractions (including the ST). The Design Documentation security assurance measure satisfies the following security assurance requirement:

- ADV_FSP.1;
- ADV_HLD.1; and,
- ADV_RCR.1.

6.2.4 Tests

The Test Documentation is found in the following documents:

- Sourcefire ISM v3.2 Test Procedures
- Sourcefire ISM v3.2 QA Test Plan
- Sourcefire ISM v3.2.3 Test Results

These documents describe the overall test plan, testing procedures, the tests themselves, including expected and actual results. In addition, these documents describe how the functional specification has been appropriately tested.

The Tests assurance measure satisfies the following assurance requirements:

- ATE_COV.1;
- ATE_FUN.1; and,
- ATE_IND.2.

6.2.5 Vulnerability Assessment

Each probabilistic or permutational mechanism used by the TOE must satisfy the SOF-Basic requirements, as required by the corresponding IDS Protection Profile. The only probabilistic or permutational mechanism is related to authentication during for login to the Management Console). Hence, FIA_UAU.1 is the only applicable security functional requirement. Sourcefire has performed a strength of function analysis that indicates that the password mechanism fulfills at least SOF-basic. Similarly, Sourcefire performed a vulnerability analysis of the TOE to identify weaknesses that can be exploited in the TOE. Both the strength of function analysis and the vulnerability analysis are documented in:

- Sourcefire Intrusion Detection System SOF for Authentication System
- Sourcefire Intrusion Management System Vulnerability Analysis

The Vulnerability Assessment assurance measure satisfies the following assurance requirements:

- AVA_SOF.1; and,
- AVA_VLA.1.

7. Protection Profile Claims

The TOE conforms to the US Government Intrusion Detection System System Protection Profile, Version 1.4, February 4, 2002.

This Security Target includes all of the assumptions and threats statements described in the PP, verbatim.

This Security Target includes all of the Security Objectives from the PP, verbatim.

This Security Target includes all of the Security Functional and Security Assurance Requirements from the PP, except those exclusively related to authenticating or communicating TSF data with external IT products. Specifically: FIA_AFL.1, FPT_ITA.1, FPT_ITC.1, and FPT_ITI.2 have been replaced by FPT_ITT.1 through the precedence of PD-0097.

Section 5 of this Security Target specifically identifies each of the operations that have been performed on requirements drawn from the PP. Note that operations already performed in the PP have not been identified in this Security Target.

The following changes have been made to requirements based on International Interpretations. These interpretations have no impact on conformance with the PP since they only serve to clarify one of the assurance claims.

- ACM_CAP.2 – per International Interpretation #3.
- ADO_IGS.1 – per International Interpretation #51.
- AVA_VLA.1 – per International Interpretation #51.

The security target includes as additional threat, T.EXPOSE. When using some Internet web browsers, it is possible to access TSF data that the user is not privileged to. This can be caused by the web browser cashing the credentials of a previously privileged user's login session. These credentials can then be used by an unprivileged user to access some TSF data when using the same web browser that the privileged user had used. This threat can be avoided if web browsers that are used to manage the TOE are properly configured to limit the space used for temporary file storage to the minimum setting, ensure that the browser uses the most current page each time it is accessed, and that it is configured to delete any temporary files when the web browser is exited. These combined configurations are the proper way to configure the web browser to ensure the strictest security for the TOE when used.

8. Rationale

This section provides the rationale for completeness and consistency of the Security Target. The rationale addresses the following areas:

- Security Objectives;
- Security Functional Requirements;
- Security Assurance Requirements;
- TOE Summary Specification;
- Security Functional Requirement Dependencies; and
- Internal Consistency.

8.1 Security Objectives Rationale

This section shows that all secure usage assumptions and organizational security policies are completely covered by security objectives. In addition, each objective counters or addresses at least one assumption or organizational security policy.

8.1.1 Security Objectives Rationale for the TOE and Environment

This section provides evidence demonstrating the coverage of organizational policies and usage assumptions by the security objectives.

	O.PROTCT	O.IDSCAN	O,IDSENS	O.IDANLZ	O,RESPON	O.EADMIN	O.ACCESS	O.IDAUTH	O.OFLOWS	O.AUDITS	O.INTEGR	O.EXPORT	O.INSTAL	O.PHYCAL	O.CREDEN	O.PERSON	O.INTROP
A.ACCESS																	X
A.DYNMIC																X	X
A.ASCOPE																	X
A.PROTCT														X			
A.LOCATE														X			
A.ITNET																	
A.MANAGE																X	
A.NOEVIL													X	X	X		
A.NOTRUST														X	X		
T.COMINT	X						X	X			X						
T.COMDIS	X						X	X				X					
T.LOSSOF	X						X	X			X						
T.NOHALT		X	X	X			X	X									
T.PRIVIL	X						X	X									
T.IMPCON						X	X	X					X				
T.INFLUX									X								
T.FACCNT										X							
T.SCNCFG		X															
T.SCNMLC		X															
T.SCNVUL		X															
T.FALACT					X												
T.FALREC				X													
T.FALASC				X													
T.MISUSE			X														
T.INADVE			X														
T.MISACT			X														
T.EXPOSE													X	X			
P.DETECT		X	X							X							
P.ANALYZ				X													
P.MANAGE	X					X	X	X					X		X	X	
P.ACCESS	X						X	X									
P.ACCACT								X		X							
P.INTGTY											X						
P.PROTCT									X					X			

Table 5 Environment to Objective Correspondence

8.1.1.1 A.ACCESS

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

The O.INTROP objective ensures the TOE has the needed access.

8.1.1.2 A.DYNMIC

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

The O.INTROP objective ensures the TOE has the proper access to the IT System. The O.PERSON objective ensures that the TOE will manage appropriately.

8.1.1.3 A.ASCOPE

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

The O.INTROP objective ensures the TOE has the necessary interactions with the IT System it monitors.

8.1.1.4 A.PROTCT

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

The O.PHYCAL provides for the physical protection of the TOE hardware and software.

8.1.1.5 A.LOCATE

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

The O.PHYCAL objective provides for the physical protection of the TOE.

8.1.1.6 A.MANAGE

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

The O.PERSON objective ensures all authorized administrators are qualified and trained to manage the TOE.

8.1.1.7 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.

The O.INSTAL objective ensures that the TOE is properly installed and operated and the O.PHYCAL objective provides for physical protection of the TOE by authorized administrators. The O.CREDEN objective supports this assumption by requiring protection of all authentication data.

8.1.1.8 A.NOTRST

The TOE can only be accessed by authorized users.

The O.PHYCAL objective provides for physical protection of the TOE to protect against unauthorized access. The O.CREDEN objective supports this assumption by requiring protection of all authentication data.

8.1.1.9 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.

The O.IDAUTH objective provides for authentication of users prior to any TOE data access. The O.ACCESS objective builds upon the O.IDAUTH objective by only permitting authorized users to access TOE data. The O.INTEGR objective ensures no TOE data will be modified. The O.PROTCT objective addresses this threat by providing TOE self-protection.

8.1.1.10 T.COMDIS

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

The O.IDAUTH objective provides for authentication of users prior to any TOE data access. The O.ACCESS objective builds upon the O.IDAUTH objective by only permitting authorized users to access TOE data. The O.PROTCT objective addresses this threat by providing TOE self-protection.

8.1.1.11 T.LOSSOF

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

The O.IDAUTH objective provides for authentication of users prior to any TOE data access. The O.ACCESS objective builds upon the O.IDAUTH objective by only permitting authorized users to access TOE data. The O.INTEGR objective ensures no TOE data will be deleted. The O.PROTCT objective addresses this threat by providing TOE self-protection.

8.1.1.12 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.

The O.IDAUTH objective provides for authentication of users prior to any TOE function accesses. The O.ACCESS objective builds upon the O.IDAUTH objective by only permitting authorized users to access TOE functions. 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.

8.1.1.13 T.PRIVIL

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

The O.IDAUTH objective provides for authentication of users prior to any TOE function accesses. The O.ACCESS objective builds upon the O.IDAUTH objective by only permitting authorized users to access TOE functions. The O.PROTCT objective addresses this threat by providing TOE self-protection.

8.1.1.14 T.IMPCON

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

The O.INSTAL objective states the authorized administrators will configure the TOE properly. The O.EADMIN objective ensures the TOE has all the necessary administrator functions to manage the product. The O.IDAUTH objective provides for authentication of users prior to any TOE function accesses. The O.ACCESS objective builds upon the O.IDAUTH objective by only permitting authorized users to access TOE functions.

8.1.1.15 T.INFLUX

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

The O.OFLOWS objective counters this threat by requiring the TOE handle data storage overflows.

8.1.1.16 T.FACCNT

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

The O.AUDITS objective counters this threat by requiring the TOE to audit attempts for data accesses and use of TOE functions.

8.1.1.17 T.SCNCFG

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

The O.IDSCAN objective counters this threat by requiring a TOE that contains a Scanner, 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.

8.1.1.18 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.

The O.IDSCAN objective counters this threat by requiring a TOE that contains a Scanner, 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.

8.1.1.19 T.SCNVUL

Vulnerabilities may exist in the IT System the TOE monitors.

The O.IDSCAN objective counters this threat by requiring a TOE that contains a Scanner, 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.

8.1.1.20 T.FALACT

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

The O.RESPON objective ensures the TOE reacts to analytical conclusions about suspected vulnerabilities or inappropriate activity.

8.1.1.21 T.FALREC

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

The O.IDANLZ objective provides the function that the TOE will recognize vulnerabilities or inappropriate activity from a data source.

8.1.1.22 T.FALASC

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

The O. IDANLZ objective provides the function that the TOE will recognize vulnerabilities or inappropriate activity from multiple data sources.

8.1.1.23 T.MISUSE

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

The O.AUDITS and O.IDSENS objectives address this threat by requiring a TOE, that contains a Sensor, collect audit and Sensor data.

8.1.1.24 T.INADVE

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

The O.AUDITS and O.IDSENS objectives address this threat by requiring a TOE, that contains a Sensor, collect audit and Sensor data.

8.1.1.25 T.MISACT

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

The O.AUDITS and O.IDSENS objectives address this threat by requiring a TOE, that contains a Sensor, collect audit and Sensor data.

8.1.1.26 T.EXPOSE

Exposure of TSF data may occur if the TOE is managed by an improperly configured web browser.

The O.INSTAL and O.PHYCAL objectives address this threat by requiring that the web browser be properly configured not to allow cached temporary files to be retained, which will avoid potential exposure.

8.1.1.27 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.

The O.AUDITS, O.IDSENS, and O.IDSCAN objectives address this policy by requiring collection of audit, Sensor, and Scanner data.

8.1.1.28 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.

The O.IDANLZ objective requires analytical processes be applied to data collected from Sensors and Scanners.

8.1.1.29 P.MANAGE

The TOE shall only be managed by authorized users.

The O.PERSON objective ensures competent administrators will manage the TOE and the O.EADMIN objective ensures there is a set of functions for administrators to use. The O.INSTAL objective supports the O.PERSON objective by ensuring administrator follow all provided documentation and maintain the security policy. The O.IDAUTH objective provides for authentication of users prior to any TOE function accesses. The O.ACCESS objective builds upon the O.IDAUTH objective by only permitting authorized users to access TOE functions. The O.CREDEN objective requires administrators to protect all authentication data. The O.PROTCT objective addresses this policy by providing TOE self-protection.

8.1.1.30 P.ACCESS

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

The O.IDAUTH objective provides for authentication of users prior to any TOE function accesses. The O.ACCESS objective builds upon the O.IDAUTH objective by only permitting authorized users to access TOE functions. The O.PROTCT objective addresses this policy by providing TOE self-protection.

8.1.1.31 P.ACCACT

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

The O.AUDITS objective implements this policy by requiring auditing of all data accesses and use of TOE functions. The O.IDAUTH objective supports this objective by ensuring each user is uniquely identified and authenticated.

8.1.1.32 P.INTGTY

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

The O.INTEGR objective ensures the protection of data from modification.

8.1.1.33 P. PROTCT

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

The O.OFLOWS objective counters this policy by requiring the TOE handle disruptions. The O.PHYCAL objective protects the TOE from unauthorized physical modifications.

8.2 Security Requirements Rationale

This section provides evidence supporting the internal consistency and completeness of the components (requirements) in the Security Target. Note that **Table 6** indicates the requirements that effectively satisfy the individual objectives.

The purpose for the environmental objectives is to provide protection for the TOE that cannot be addressed through IT measures. The defined objectives provide for physical protection of the TOE, proper management of the TOE, and interoperability requirements on the TOE. Together with the IT security objectives, these environmental objectives provide a complete description of the responsibilities of TOE in meeting security needs.

All of the SFRs have been derived from the IDSSPP. All operations completed in this Security Target have been completed in accordance with the IDSSPP. The only other SFR-related changes involve the omission of three SFRs related to communication with other (trusted) IT products. Since the Sourcefire products are not intended to interact with other IT products, these requirements are not relevant and can be omitted without impacting the fulfillment of the security objectives in the IDSSPP. Furthermore, the SFRs that have been omitted are not relied upon by other SFRs (see Section 8.4). Ultimately, completion of operations as allowed by the IDSSPP and removal of SFRs that are justifiably unnecessary cannot impact consistency among the SFRs and omission of SFRs that are not required by other SFRs or the security objectives cannot impact the mutual support among the SFRs.

8.2.1 Security Functional Requirements Rationale

All Security Functional Requirements (SFR) identified in this Security Target are fully addressed in this section and each SFR is mapped to the objective for which it is intended to satisfy.

	O.PROTCT	O.IDSCAN	O,IDSENS	O.IDANLZ	O,RESPON	O.EADMIN	O.ACCESS	O.IDAUTH	O.OFLOWS	O.AUDITS	O.INTEGR	O.EXPORT
FAU_GEN.1										X		
FAU_SAR 1						X						
FAU_SAR.2							X	X				
FAU_SAR.3						X						
FAU_SEL.1						X				X		
FAU_STG.2	X						X	X	X		X	

	O.PROTCT	O.IDSCAN	O,IDSENS	O.IDANLZ	O,RESPON	O.EADMIN	O.ACCESS	O.IDAUTH	O.OFLOWS	O.AUDITS	O.INTEGR	O.EXPORT
FAU_STG.4									X	X		
FIA_UAU.1							X	X				
FIA_ATD.1								X				
FIA_UID.1							X	X				
FMT_MOF.1	X						X	X				
FMT_MTD.1	X						X	X			X	
FMT_SMR.1								X				
FPT_ITT.1											X	X
FPT_RVM.1	X					X		X		X	X	
FPT_SEP.1	X					X		X		X	X	
FPT_STM.1										X		
IDS_SDC.1		X	X									
IDS_ANL.1				X								
IDS_RCT.1					X							
IDS_RDR.1						X	X	X				
IDS_STG.1	X						X	X	X		X	
IDS_STG.2									X			

Table 6 Objective to Requirement Correspondence

8.2.1.1 O.PROTCT

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

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 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 ensure that all functions are invoked and succeed before each function may proceed [FPT_RVM.1]. The TSF must be protected from interference that would prevent it from performing its functions [FPT_SEP.1].

8.2.1.2 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.

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].

8.2.1.3 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.

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].

8.2.1.4 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 Analyzer is required to perform intrusion analysis and generate conclusions [IDS_ANL.1].

8.2.1.5 O.RESPON

> *The TOE must respond appropriately to analytical conclusions.*

The TOE is required to respond accordingly in the event an intrusion is detected [IDS_RCT.1].

8.2.1.6 O.EADMIN

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

The TOE must provide the ability to review and manage the audit trail of the System [FAU_SAR.1, FAU_SAR.3, 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 ensure that all functions are invoked and succeed before each function may proceed [FPT_RVM.1]. The TSF must be protected from interference that would prevent it from performing its functions [FPT_SEP.1].

8.2.1.7 O.ACCESS

> *The TOE must allow authorized users to access only appropriate TOE functions and data.*

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, 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]. 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].

8.2.1.8 O.IDAUTH

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

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, 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]. 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 [FPT_RVM.1]. The TSF must be protected from interference that would prevent it from performing its functions [FPT_SEP.1].

8.2.1.9 O.OFLOWS

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

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 the 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 the its audit trail is full [IDS_STG.2].

8.2.1.10 O.AUDITS

The TOE must record audit records for data accesses and use of the System 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 the its audit trail is full [FAU_STG.4]. The TOE must ensure that all functions are invoked and succeed before each function may proceed [FPT_RVM.1]. The TSF must be protected form interference that would prevent it from performing its functions [FPT_SEP.1]. Time stamps associated with an audit record must be reliable [FPT_STM.1].

8.2.1.11 O.INTEGR

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

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 System must protect the collected data from modification and ensure its integrity when the data is transmitted to another part of the TOE [FPT_ITT.1]. The TOE must ensure that all functions to protect the data are not bypassed [FPT_RVM.1]. The TSF must be protected form interference that would prevent it from performing its functions [FPT_SEP.1].

8.2.1.12 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 TOE must protect all data from modification and ensure its integrity when the data is transmitted to another TOE component [FPT_ITT.1].

8.3 Security Assurance Requirements Rationale

This ST contains the assurance requirements from the CC EAL2 assurance package and is based on good commercial development practices to provide a low to moderate level of assurance . While the System may monitor a hostile environment, it is expected to be in a non-hostile position and embedded in or protected by other products designed to address threats that correspond with the intended environment. Note that the security environment assumes physical protection and the TOE itself offers only a very limited interface and can only be configured during initialization, offering essentially no opportunity for an attacker to subvert the security policies without physical access. As such, it is believed that EAL 2 provides an appropriate level of assurance in the security functions offered by the TOE.

8.4 Requirement Dependency Rationale

The ST satisfies all the requirement dependencies of the Common Criteria, except as noted below. Table 7 Requirement Dependency Rationale lists each requirement from Section 5.1 with a dependency and indicates which requirement was included to satisfy the dependency, if any. For each dependency not included, a justification is proved.

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 and 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_SMF.1	NO*
FMT_MTD.1	FMT_SMR.1	YES
	FMT_SMF.1	NO*
FMT_SMR.1	FIA_UID.1	YES

Table 7 Requirement Dependency Rationales

* Prior to the publication and verification of the IDS System PP, International Interpretation #65 was finalized. This interpretation introduced a new family of Security Management requirements, Specification of Management Functions (FMT_SMF). While this should not normally affect dependency rationale, that interpretation introduces dependencies from FMT_MOF.1 and FMT_MTD.1, both contained in this Security Target. Hence, it seems as though some FMT_MSA security requirements should be added to this Security Target to fulfill those dependencies. However, while the IDS System PP is clearly intended to ensure that certain security management functions are controlled if they are made available, it is not evident from the IDS System PP which, if any, of those security management functions must be present in the first place. This Security Target identifies all applicable security management functions in the TOE and explains how they are appropriately controlled and it is effectively unnecessary to introduce a security functional requirement to demand that certain security management functions must be present.

8.5 Explicitly Stated Requirements Rationale

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 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.

8.6 TOE Summary Specification Rationale

Each subsection in Section 6, the TOE Summary Specification, describes a security function of the TOE. Each description is followed with rationale that indicates which requirements are satisfied by aspects of the corresponding security function. Working together, this set of security functions satisfy all of the security functions and assurance requirements. Furthermore, all of the security functions are necessary in order for the TSF to provide the required security functionality.

This Section in conjunction with Section 6, the TOE Summary Specification, provides evidence that the security functions are suitable to meet the TOE security requirements. The c security functions work together to provide all of the security requirements. The security functions described in the TOE summary specification are all necessary for the required security functionality in the TSF. **Table 8 Security Functions vs. Requirements Mapping** demonstrates the relationship between security requirements and security functions.

	SECURITY AUDIT	IDENTITY & AUTHENTICATION	SECURITY MANAGEMENT	PROTECTION OF SECURITY FUNCTIONS	SYSTEM DATA COLLECTION	SYSTEM DATA ANALYSIS	SYSTEM DATA REVIEW, AVAILABILITY & LOSS
FAU_GEN.1	X						
FAU_SAR 1	X						
FAU_SAR.2	X						
FAU_SAR.3	X						
FAU_SEL.1	X						
FAU_STG.2	X						
FAU_STG.4	X						
FIA_UAU.1		X					
FIA_ATD.1		X					
FIA_UID.1		X					
FMT_MOF.1			X				
FMT_MTD.1			X				
FMT_SMR.1			X				
FPT_ITT.1				X			
FPT_RVM.1				X			
FPT_SEP.1				X			
FPT_STM.1				X			
IDS_SDC.1					X		
IDS_ANL.1						X	
IDS_RCT.1						X	
IDS_RDR.1							X
IDS_STG.1							X
IDS_STG.2							X

Table 8 Security Functions vs. Requirements Mapping

8.7 PP Claims Rationale

See section 7, Protection Profile Claims.

8.8 Strength of Function Rationale

The TOE minimum strength of function is SOF-basic. The 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 4.