symantec™

CyberWolf v2.0
Security Target

Version 1.0

Prepared by:

Symantec Corporation
20300 Stevens Creek Boulevard
Cupertino, California 95014

Computer Sciences Corporation
132 National Business Parkway
Annapolis Junction, Maryland 20701

Date	Revision	Changes Made
September 12, 2003	1.1	Original Draft
October 21, 2003 thru November 14, 2003	1.2 – 1.14	ST in internal development process
December 19, 2003	1.16	Grammatical and Clarification corrections made
January 20, 2004	1.17	Corrections in response to EDR 3
January 20, 2004	1.18	Correction made to threat T.ALTER_CONFIG
January 22, 2004	1.19	Revisions made to TSF_DRE.
February 20, 2004	1.20	Added statement about using single monitor.
April 26, 2004	1.21	Corrections in response to EDR 20
April 26, 2004	1.22	Made this document version 1.0

Table of Contents

List of Figures

List of Tables

1 SECURITY TARGET INTRODUCTION

1 This Chapter presents security target (ST) identification information and an overview of the ST. An ST contains the information technology (IT) security requirements of an identified Target of Evaluation (TOE) and specifies the functional and assurance security measures offered by that TOE to meet stated requirements. An ST principally defines:

a) A security problem expressed as a set of assumptions about the security aspects of the environment, a list of threats that the product is intended to counter, and any known rules with which the product must comply (Chapter 3, TOE Security Environment).

b) A set of security objectives and a set of security requirements to address the security problem (Chapters 4 and 5, Security Objectives and IT Security Requirements, respectively).

c) The IT security functions provided by the TOE that meet the set of requirements (Chapter 6, TOE Summary Specification).

2 The structure and content of this ST comply with the requirements specified in the Common Criteria (CC), Part 1, Annex C, and Part 3, Chapter 5.

1.1 ST and TOE Identification

3 This section provides information needed to identify and control this ST and its Target of Evaluation (TOE). This ST targets Evaluation Assurance Level (EAL)2.

ST Title:	Symantec CyberWolf v2.0 Security Target
ST Version:	1.0
Revision Number:	$Revision: 1.22 $
Publication Date:	$Date: 2004/04/26 11:42:52 $
Authors:	Computer Sciences Corporation
TOE Identification:	Symantec CyberWolf.v2.0
CC Identification:	Common Criteria for Information Technology Security Evaluation, Version 2.1, August 1999 (also known as ISO 15408)
ST Evaluator:	Computer Sciences Corporation Common Criteria Testing Laboratory
Keywords:	Automated Incident Response

1.2 References

4 The following documentation was used to prepare this ST:

[CC_PART1] Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and general model, dated August 1999, version

	2.1, CCIMB-99-031, Incorporated with interpretations as of 2002-02-28
[CC_PART2]	Common Criteria for Information Technology Security Evaluation – Part 2: Security functional requirements, dated August 1999, version 2.1, CCIMB-99-032 , Incorporated with interpretations as of 2002-02-28.
[CC_PART3]	Common Criteria for Information Technology Security Evaluation – Part 3: Security assurance requirements, dated August 1999, version 2.1, CCIMB-99-033, Incorporated with interpretations as of 2002-02-28.
[CEM_PART1]	Common Evaluation Methodology for Information Technology Security – Part 1: Introduction and General Model, dated 1 November 1997, version 0.6.
[CEM_PART2]	Common Methodology for Information Technology Security Evaluation – Part 2: Evaluation Methodology, dated August 1999, version 1.0.
[JSS_V23]	Java Servlet Specification 2.3, Java Software
[SFP_V12]	JavaServer Pages 1.2 Specification, Java Software

1.3 Conventions, Terminology, and Acronyms

5 This section identifies the formatting conventions used to convey additional information and terminology. It also defines terminology and the meanings of acronyms used throughout this ST.

1.3.1 Conventions

6 This section describes the conventions used to denote Common Criteria (CC) operations on security functional components and to distinguish text with special meaning. The notation, formatting, and conventions used in this ST are largely consistent with those used in the CC. Selected presentation choices are discussed here.

7 The CC allows several operations to be performed on security functional components; *assignment, refinement*, *selection,* and *iteration* as defined in paragraph 2.1.4 of Part 2 of the CC are:

a) The *assignment* operation is used to assign a specific value to an unspecified parameter, such as the length of a password. Showing the value in square brackets [assignment_value(s)] indicates an assignment.

b) The *refinement* operation is used to add detail to a requirement, and thus further restricts a requirement. Refinement of security requirements is denoted by **bold text**.

c) The *selection* operation is used to select one or more options provided by the CC in stating a requirement. Selections are denoted by *underlined italicized text.*

d) *Iterated* functional components are given unique identifiers by appending to the component name, short name, and functional element name from the CC an iteration number inside parenthesis, i.e., FMT_MTD.1.1 (1) and FMT_MTD.1.1 (2).

e) Plain *italicized text* is used to emphasize text.

1.3.2 Terminology

8 In the CC, many terms are defined in Section 2.3 of Part 1. The following terms are a subset of those definitions:

Authentication data	Information used to verify the claimed identity of a user.
Authorized User	A user who may, in accordance with the TOE Security Policy (TSP[1]), perform an operation.
Component	For the purpose of this document, a component is an individual CyberWolf process. The CyberWolf components are: Manager, Monitor, SecurSite, and each Device Expert. There is a slight distinction between component and subsystem since Tomcat will be considered a subsystem, but will not be defined as a component as it is simply an execution environment and webserver.
External IT entity	Any IT product or system, untrusted or trusted, outside of the TOE that interacts with the TOE.
Human user	Any person who interacts with the TOE.
Identity	A representation (e.g., a string) uniquely identifying an authorized user, which can either be the full or abbreviated name of that user or a pseudonym.
InfoManager ***Information Manager***	The TOE 'Manager' subsystem may be referred throughout documentation as 'Manager', 'InfoManager', or 'Information Manager'
ISS RS Expert	The CyberWolf Device Expert which runs on ISS RealSecure systems to collect alert data. This device expert is a TOE subsystem.
Object	An entity within the TOE Security Function (TSF[2]) Scope of Control (TSC[3]) that contains or receives information and upon which subjects perform operations.
Role	A predefined set of rules establishing the allowed interactions between a user and the TOE.
Security Functional Components	Express security requirements intended to counter threats in the assumed operating environment of the TOE.

1 TSP – A set of rules that regulate how assets are managed, protected and distributed within a TOE.
As defined in the CC, Part 1, version 2.1:
2 TSF - A set consisting of all hardware, software, and firmware of the TOE that must be relied upon for the correct enforcement of the TSP.
3 TSC - The set of interactions that can occur with or within a TOE and are subject to the rules of the TSP.

Snort Expert	The CyberWolf Device Expert which runs on Snort systems to collect alert data. This device expert is a TOE subsystem.
Subject	An entity within the TSC that causes operations to be performed.
User	Any entity (human user or external IT entity) outside the TOE that interacts with the TOE.

9 The following terminology is specific to this ST.

Administrator	An authorized user who manages the CyberWolf product.
Junior Incident Handler	An authorized user who responds to CyberWolf incidents with a set of security management functions defined in Section 5.
Senior Incident Handler	An authorized user who responds to CyberWolf incidents with a set of security management functions defined in Section 5.
Read only user	An authorized user who can only read, but not alter CyberWolf collected and generated data.

1.3.3 Acronyms

10 The following acronyms are used in this Security Target:

ACRONYM	DEFINITION
AUT	Authentication
CC	Common Criteria for Information Technology Security Evaluation
CEM	Common Evaluation Methodology for Information Technology Security
CM	Configuration Management
EAL	Evaluation Assurance Level
FDP	User Data Protection CC Class
FIA	Identification and Authentication CC Class
FMT	Security Management CC Class
FPT	Protection of Security Functions
FSP	Functional Specification
HLD	High Level Design
ISO	International Standards Organization
ISO 15408	Common Criteria 2.1 ISO Standard
IT	Information Technology
JSP	JavaServer Pages
MOF	Management of Functions
MTD	Management of TSF Data
OSP	Organization Security Policy
PP	Protection Profile
SAR	Security Assurance Requirement
SFP	Security Function Policy
SFR	Security Functional Requirement

ACRONYM	DEFINITION
SM	Security Management
SMR	Security Management Roles
SOF	Strength of Function
ST	Security Target
TOE	Target of Evaluation
TSC	TSF Scope of Control
TSF	TOE Security Function
TSP	TOE Security Policy
UAU	User Authentication
UDP	User Data Protection

1.4 TOE Overview

11 The TOE is an automated incident reporting system designed for security operations centers (SOCs) and managed security service providers (MSSPs) that need automated incident reports in near real-time. CyberWolf provides correlation of high volumes of security alert information generated by computers, network devices and intrusion detection sensors. CyberWolf automates the detection and analysis of events and alerts to define security incidents. By analyzing the thousands of alerts most likely present during an attack, CyberWolf tracks and correlates these alerts to determine attack patterns and the development of a security incident.

12 A summary of the TOE security functions can be found in Section 2, TOE Description. A detailed description of the security functions can be found in Section 6, TOE Summary Specification.

1.5 Common Criteria Conformance Claim

13 This ST conforms to CC Part 2 extended, and is CC Part 3 conformant at the EAL 2 level of assurance.

2 TOE DESCRIPTION

14 This section provides context for the TOE evaluation by identifying the product type and describing the evaluated configuration.

2.1 Product Type

15 CyberWolf v2.0 is an automated incident management system designed for security operations centers (SOCs) and managed security service providers (MSSPs) that need automated incident management in near real-time. CyberWolf provides correlation of high volumes of security alert information generated by computers, network devices and intrusion detection sensors. CyberWolf automates the detection and analysis of events and alerts to define security incidents. By analyzing the thousands of alerts most likely present during an attack, CyberWolf tracks and correlates these alerts to determine attack patterns and the development of a security incident.

16 CyberWolf v2.0 is an application of the Apache's Jakarta **Tomcat** project. Tomcat is Sun's official reference implementation of the Servlet and JSP specifications. Tomcat is developed in an open and participatory environment and released under the Apache Software License. For the TOE, Tomcat is included to facilitate the web browser interface. Table 1 lists the software evaluated, as well as associated hardware and software components not evaluated.

Table 1: Evaluated Components

	Description	Version
Evaluated Software	CyberWolf	2.0
	Tomcat	4.06
Un-evaluated Software	Oracle	8/9
	Solaris - or -	7 or better
	Windows NT/2000/XP	
	MS IE	5.5 SP2 or above 6.0
Un-evaluated Hardware	SUN SPARC or 512 MB RAM	
CyberWolf	Pentium PC Compatible	
	1 Ghz Processor, 512 MB RAM	
Device experts	200 Mhz Processor, 64 MB RAM	

2.1.1 Physical Scope and Boundary

17 Figure 1 shows a basic environment for the TOE. In the evaluated configuration, CyberWolf receives alert and security incident information from CyberWolf Device Experts residing on systems running Internet Security Systems' RealSecure™ and/or Snort™, the open source network intrusion detection system. The TOE consists of the CyberWolf system (SecurSite, Tomcat, Monitor, and Manager subsystems) running on one host, the ISS RealSecure Expert which runs on a RealSecure system, and the Snort Expert which runs on a Snort system. CyberWolf stores user names, passwords, alerts and security incident related information in an Oracle 8/9 database. For the purposes of the TOE evaluation, Oracle runs on the same host running the CyberWolf System. The Oracle database is not considered part of the TOE and is assumed to operate correctly and securely. Users and CyberWolf administrators operate the TOE via a web browser (java is not needed). Although the CyberWolf systems can operate with multiple Monitors, the evaluated configuration of the TOE is the CyberWolf system using a single Monitor.

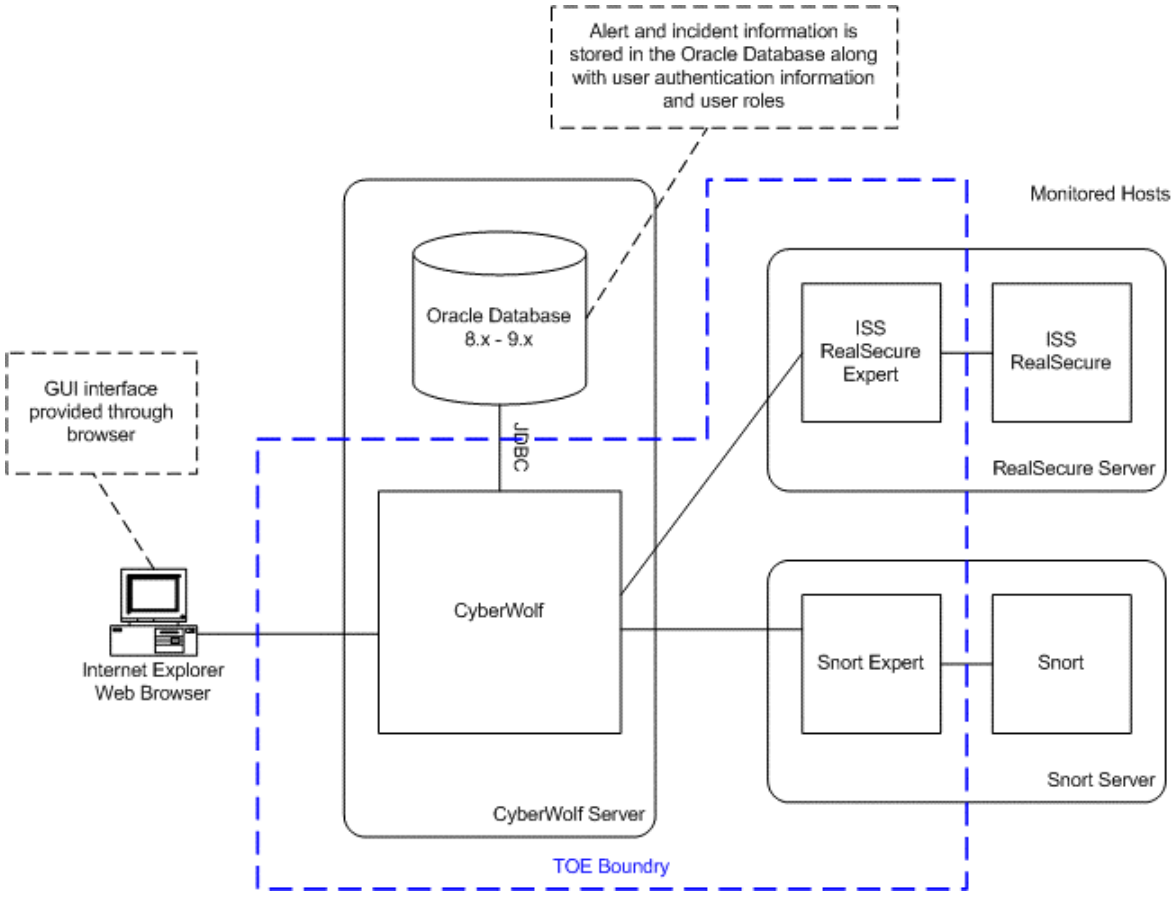

Figure 1: Physical Scope and Boundary

2.1.2 Logical Scope and Boundary

18 CyberWolf v2.0 uses a client-server architecture consisting of CyberWolf *Device Experts* and the CyberWolf Manager. The CyberWolf SecureSite, for providing web access, is built on top of the Apache Tomcat v4.06 and depends upon Tomcat for its execution environment. The Tomcat subsystem receives HTTPS web requests and replies by compiling and executing a corresponding JSP (code which is part of the SecurSite subsystem). The resulting HTML is sent back to the web browser which made the request and the JSP remains in memory as a servlet until Tomcat is restarted or memory needs to be reclaimed. This allows Tomcat to simply execute each JSP as needed without recompilation when pages are requested more than once. The CyberWolf Manager and Monitor are Java applications that require the Java Runtime Environment to run. However, Java is not the focus of this security target and is assumed to function according to its provided specifications.

19 The TOE logical boundary is the following security functions controlled by the TOE:

- Identification and Authentication (TSF_INA)
- Security Management (TSF_FMT)
- User Action Log (TSF_UAL)
- Data Collection (TSF_EDC)
- Key Management (TSF_KMG)
- Communications Security (TSF_CCS)
- Data Reporting (TSF_DRE)

20 **Identification and Authentication (TSF_INA):** CyberWolf's I&A mechanism is built on top of Tomcat's Java Database Connectivity (JDBC) Realms. CyberWolf stores username and the MD5 hash of the user's password in a table in the Oracle database. Passwords are encrypted using the MD5 algorithm. To perform identification and authentication, SecurSite and Tomcat verify that the supplied user name exists in the system by checking for the presence of that user name in the Oracle database then extract the MD5 hash of that user's password from the Oracle database and compare it with a hash of the user entered password.

21 **Security Management (TSF_FMT):** CyberWolf differentiates between four user roles. Each user role is assigned a limited number of security functions that the role can perform on the TOE. The roles defined for TOE usage are: Administrator, Senior Incident Handler, Junior Incident Handler, Read Only User.

22 **User Action Log (TSF_UAL):** CyberWolf collects a log of certain user actions that result in changes to the Oracle database. The logs include the user name performing the action, the type of event, the date and time of the event, and the outcome of the event. The log also includes any additional information that may be pertinent.

23 **Data Collection (TSF_EDC):** CyberWolf utilizes its Device Experts to collect data from security components outside the TOE. The device experts first collect data using sensors custom made to the security component being monitored. The TOE contains The RealSecure Device Expert which is programmed to read from the RealSecure database and the Snort Device Expert

which reads from the Snort log file. The data is then translated into a common form that is recognized throughout CyberWolf components. Event data is sent back to the Manager.

24 **Key Management (TSF_KMG):** CyberWolf performs Key Management through the use of the Monitor subsystem. The Monitor maintains a list of each of the active keys and their associated components. As each component starts, it generates its own symmetric secret key (with the maximum bit size supported by the selected encryption algorithm). Between every one and three hours, each component also randomly generates a new secret key. The Monitor subsystem performs key management for the Manager, SecurSite, and each Device Expert subsystem. Key Management is unnecessary for the Tomcat subsystem since Tomcat is simply an execution environment.

25 Every time a connection is re-established with the Monitor, a component starts, or a component generates a new key, the CyberWolf component registers its current key with the Monitor. For inter-component communication in CyberWolf, a component requests connection information from the Monitor for the targeted component.

26 **Communications Security (TSF_CCS):** All message traffic between CyberWolf components is encrypted. At the time CyberWolf is installed, the type of encryption can be selected. The administrator can select DES, TripleDES, or Blowfish. When the Monitor component is installed, it generates a shared secret key of maximum length. This secret key is then encrypted with a pseudo-random password. These keys are stored in the file system in two files. In order for any other component to communicate with the Monitor (which is necessary to communicate with any other CyberWolf component), these two key files must be manually copied to that component's system. The key is then used as an encryption key for sending messages to the CyberWolf Monitor. The CyberWolf Monitor uses it as its decryption key.

27 **Data Reporting (TSF_DRE):** CyberWolf data reporting is automatic. Reporting is done in both real-time for listing alerts and incidents and generated graphically in predefined intervals. Reports are viewable by all valid CyberWolf users. Reports are generated on a daily and weekly basis. By default daily reports are generated at 2:00 AM everyday and weekly reports are generated on every Wednesday at the same time specified for the daily reports (i.e. 2:00 AM). Users must manually edit the reports configuration file if the desired reporting schedule does not match the default settings. CyberWolf reports are generated directly from the data in the database at the time the report is run.

28 Real-time reporting of Incidents and Alerts are also available in a browsable, web interface which provides up to the minute displays of the incidents and alerts received by the system each time the page is loaded. Incident and Alert lists can be filtered and sorted as necessary and users can add conclusions, notes, and actions to each incident.

3 TOE SECURITY ENVIRONMENT

3.1 Secure Usage Assumptions

29 This section describes the security aspects of the intended environment for the evaluated TOE. This includes information about the physical, personnel, procedural, connectivity, and functional aspects of the environment.

30 The operational environment must be managed in accordance with assurance requirement documentation for delivery, operation, and user/system administrator guidance. The following specific conditions are assumed to exist in an environment where this TOE is employed.

3.1.1 Environment Assumptions

31 The environmental assumptions delineated in Table 2 are required to ensure the security of the TOE:

Table 2: Environmental Assumptions

Assumption	Description
A.INSTALL	The TOE has been delivered, installed, and setup in accordance with documented delivery and installation/setup procedures.
A.MANAGE	There will be one or more competent system administrator(s) assigned to manage the TOE and the security of the information it contains.
A.NO_EVIL_ADM	The system administrator(s) are not careless, willfully negligent, nor hostile, and will follow and abide by the instructions provided by the system administration documentation.
A.PROCEDURE	Procedures exist for granting system administrator(s) access to the TSF.
A.CHANGE_PWD	Users and administrators change their passwords every 60 days
A.PHYSICAL_ PROTECT	The TOE will be located within facilities providing controlled access to prevent unauthorized physical access.
A.RELIABLE_TIME	The host machines running the TOE software will provide the TOE with a reliable time and date.
A.ACCESS_CONTROL	The operating systems upon which the TOE software runs will be configured to restrict modification to TOE executables, configuration files, and cryptographic keys to only the CyberWolf authorized administrators.

3.2 Threats

32 Table 3 identifies the threats to the TOE. The threats to the TOE are considered to be users with public knowledge of how the TOE operates. However, the threats do not possess access to the resources necessary to perform a cryptanalysis on the algorithms used. The threat has access to the TOE. Mitigation to the threats is through the objectives identified in Section 4, Security Objectives.

Table 3: Threats to the TOE

Threat	Description
T.ALTER_CONFIG	An unauthorized user may attempt to access the TOE through an external interface in order to alter the TOE configuration to circumvent the configured policy so they can obscure intrusion attempts on the network from the TOE's users.

3.3 Organizational Security Policies

33 An organizational security policy is a set of rules, practices, and procedures imposed by an organization to address its security needs. Table 4 identifies the organizational security policies applicable to the TOE.

Table 4: Applicable Organization Security Policies

P.ADMIN	Management functions of the TOE shall be restricted to the Authorized Administrator
P.ACCACT	Human users of the TOE shall be accountable for their actions.
P.REPORT	Reports on network activities will be made based on collected event data.
P.DATA_COLLECT	The TOE shall collect event data from supported IT security products.

4 SECURITY OBJECTIVES

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

- Security objectives for the TOE, and
- Security objectives for the environment.

4.1 Security Objectives for the TOE

35 This section identifies and describes the security objectives of the TOE.

36 The TOE accomplishes the security objectives defined in Table 5.

Table 5: Security Objectives for the TOE

Objectives	Description
O.COLLECT	The TOE must collect event data from ISS RealSecure and Snort IDS.
O.REPORT	The TOE must report event data based on TOE policy.
O.ADMIN	The TOE must include a set of functions that allow management of its functions and data.
O.SEP_ROLE	The TOE must accommodate separate roles for Authorized Administrators to limit their access to the TOE security mechanisms
O.CONF_DATA	The TOE will keep confidential all data that is sent between components of the TOE.
O.LOGGING	The TOE will maintain a log of a security-relevant subset of user actions.

4.2 Security Objectives for the Environment

37 The security objectives for the IT Environment are defined in Table 6.

Table 6: Security Objectives for the TOE Environment

Objectives	Description
OE.DAC	The TOE environment must provide discretionary access control (DAC) to protect TOE executables, TOE data, and host generated data.
OE.PLATFORM_SUPPORT	The TOE environment must provide reliable platform functions including: correct hardware operation and functionality including providing system time; correct platform software operation and functionality.
OE.PHYSICAL	Those responsible for the TOE must ensure that those parts of the TOE critical to security policy are protected from any physical attack.

Objectives	Description
OE.INSTALL	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.

5 IT SECURITY REQUIREMENTS

38 This section defines the IT security requirements that shall be satisfied by the TOE or its environment:

39 The CC divides TOE security requirements into two categories:

- Security functional requirements (SFRs) (such as, identification and authentication, security management, and user data protection) that the TOE and the supporting evidence need to satisfy to meet the security objectives of the TOE.

- Security assurance requirements (SARs) that provide grounds for confidence that the TOE and its supporting IT environment meet its security objectives (e.g., configuration management, testing, and vulnerability assessment).

40 These requirements are discussed separately within the following subsections.

5.1 TOE Security Functional Requirements

41 The TOE satisfies the SFRs delineated in Table 7. The rest of this section contains a description of each component and any related dependencies.

Table 7: TOE Security Functional Requirements

Class FIA: Identification and Authentication	
FIA_UAU.2	User Authentication Before Any Action
FIA_UAU.7	Protected Authentication Feedback
FIA_UID.2	User identification before any action
Class FMT: Security Management	
FMT_MOF.1	Management of Security Functions Behavior
FMT_SMF.1	Specification of Management Functions
FMT_SMR.1	Security roles
Class FPT: Protection of the TSF	
FPT_ITT.1	Basic internal TSF data transfer protection
Class FCS: Cryptographic Support	
FCS_CKM.1	Cryptographic key generation
FCS_CKM.2	Cryptographic key distribution
FCS_CKM.4	Cryptographic key destruction
FCS_COP.1	Cryptographic operation

5.1.1 Class FCS: Cryptographic Support

42 FCS_CKM.1 Cryptographic key generation

Hierarchical to: No other components.

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [Symantec CyberWolf Proprietary Key Generation Algorithm] and specified cryptographic key sizes [56 bits for DES, 168 bits for 3DES, 448 bits for Blowfish] that meet the following: [none].

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

FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

43 FCS_CKM.2 Cryptographic key distribution

Hierarchical to: No other components.

FCS_CKM.2.1 The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method [CyberWolf key distribution method] that meets the following: [none].

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

FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

44 FCS_CKM.4 Cryptographic key destruction

Hierarchical to: No other components.

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [overwrite] that meets the following: [none].

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

FMT_MSA.2 Secure security attributes

45 FCS_COP.1 Cryptographic operation

Hierarchical to: No other components.

FCS_COP.1.1 The TSF shall perform [data encryption and decryption] in accordance with a specified cryptographic algorithm [DES, 3DES or Blowfish] and cryptographic key sizes [56 bits for DES, 168 bits

for 3DES, 448 bits for Blowfish] that meet the following: [FIPS PUB 46-3 (DES and 3DES) none (Blowfish[4]).]

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

FCS_CKM.4 Cryptographic key destruction
FMT_MSA.2 Secure security attributes

5.1.2 Class FIA: Identification and Authentication

46 FIA_UAU.2 User Authentication Before Any Action

Hierarchical to: FIA_UAU.1 Timing of Authentication

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

Dependencies: FIA_UID.1 Timing of Identification

47 FIA_UAU.7 Protected Authentication Feedback

Hierarchical to: No other components

FIA_UAU.7.1 The TSF shall provide only [obscured feedback] to the user while the authentication is in progress.

Dependencies: FIA_UAU.1 Timing of Authentication

48 FIA_UID.2 User identification before any action

Hierarchical to: FIA_UID.1

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

Dependencies: No dependencies

5.1.3 Class FMT: Security Management

49 FMT_MOF.1(1) Management of Security Functions Behavior

Hierarchical to: No other components

[4] While a formal standard for Blowfish has not been published, the official web site for Blowfish is http://www.schneier.com/blowfish.html. This web site contains test vectors and a reference implementation.

	FMT_MOF.1.1	The TSF shall restrict the ability to *enable, disable, modify the behavior of* the functions [User management, device expert communication, view server information] to [the Administrator].
	Dependencies:	FMT_SMR.1 Security Roles FMT_SMF.1 Specification of management functions

50 FMT_MOF.1(2) Management of Security Functions Behavior

	Hierarchical to:	No other components
	FMT_MOF.1.1	The TSF shall restrict the ability to *disable and enable* the functions [create incidents, assign any incident, modify any incident, unassign any incident, close any incident] to [the Administrator, the Senior Incident Handler].
	Dependencies:	FMT_SMR.1 Security Roles FMT_SMF.1 Specification of management functions

51 FMT_MOF.1(3) Management of Security Functions Behavior

	Hierarchical to:	No other components
	FMT_MOF.1.1	The TSF shall restrict the ability to *disable and enable* the functions [assign incidents assigned to them, modify incidents assigned to them, close incidents assigned to them] to [the Administrator, the Senior Incident Handler, the Junior Incident Handler].
	Dependencies:	FMT_SMR.1 Security Roles FMT_SMF.1 Specification of management functions

52 FMT_MOF.1(4) Management of Security Functions Behavior

	Hierarchical to:	No other components
	FMT_MOF.1.1	The TSF shall restrict the ability to *disable and enable* the functions [view CyberWolf reports, Query CyberWolf alerts, View CyberWolf incidents]

to [the Administrator, the Senior Incident Handler, the Junior Incident Handler, the Read-only User].

Dependencies: FMT_SMR.1 Security Roles
FMT_SMF.1 Specification of management functions

53 FMT_SMF.1 Specification of Management Functions

Hierarchical to: No other components.

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

User management,
device expert communication,
view server information,
create incidents,
assign any incident,
modify any incident,
unassign any incident,
close any incident,
assign incidents assigned to them,
modify incidents assigned to them,
close incidents assigned to them,
view CyberWolf reports,
Query CyberWolf alerts,
View CyberWolf incidents].

Dependencies: No Dependencies

54 FMT_SMR.1 Security roles

Hierarchical to: No other components.

FMT_SMR.1.1 The TSF shall maintain the roles [

- Administrator
- Senior Incident Handler
- Junior Incident Handler
- Read-only User]

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

Dependencies: FIA_UID.1 Timing of identification

5.1.4 Class FPT: Protection of the TSF

55 FPT_ITT.1 Basic internal TSF data transfer protection

Hierarchical to: No other components.

FPT_ITT.1.1 The TSF shall protect TSF data from *disclosure* when it is transmitted between separate parts of the TOE.

5.2 TOE Security Assurance Requirements

56 Table 8 identifies the security assurance components drawn from CC Part 3 Security Assurance Requirements EAL2. The SARs are not iterated or refined from Part 3.

Table 8: EAL2 Assurance Requirements

Assurance Component ID	Assurance Component Name	Dependencies
ACM_CAP.2	Configuration items	None
ADO_DEL.1	Delivery procedures	None
ADO_IGS.1	Installation, generation, and start-up procedures	AGD_ADM.1
ADV_FSP.1	Informal functional specification	ADV_RCR.1
ADV_HLD.1	Descriptive high-level design	ADV_FSP.1, ADV_RCR.1
ADV_RCR.1	Informal correspondence demonstration	None
AGD_ADM.1	Administrator guidance	ADV_FSP.1
AGD_USR.1	User guidance	ADV_FSP.1
ATE_COV.1	Evidence of coverage	ADV_FSP.1, ATE_FUN.1
ATE_FUN.1	Functional testing	None
ATE_IND.2	Independent testing-sample	ADV_FSP.1, AGD_ADM.1, AGD_USR.1, ATE_FUN.1
AVA_SOF.1	Strength of TOE security function evaluation	ADV_FSP.1, ADV_HLD.1
AVA_VLA.1	Developer vulnerability analysis	ADV_FSP.1, ADV_HLD.1 AGD_ADM.1, AGD_USR.1

5.3 Security Requirements for the IT Environment

57 There are no security functional requirements for the IT Environment.

5.4 Explicitly Stated Requirements for the TOE

58 SCW_UAL.1 User Action Log

Hierarchical to: No other components

SCW_UAL.1.1 The TSF shall be able to generate a record of the following events:

Event	Description
Add Action to Incident	The users enters an action description to an incident in the database.
Add Conclusion to Incident	The user recorded a conclusion for the incident
Add Incident	The user manually adds a new incident to the database
Add Users	The user added a new user to the TOE
Assign Incident to Other Users	The user assigned an incident to another user, so that user can record a conclusion about it
Assign Incident to Self	The user assigned an incident to the user, so they can record a conclusion about it
Change Incident Priority	The user increased or decreased recorded priority of an incident
Change Incident Status(Open, Closed, Working)	The user changed the status of an incident
Update Tracking Rule	The user updated a tracking rule for the incident

(EXP)

SCW_UAL.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) the additional information as described below:

Event	Additional information recorded
Add Action to Incident	Incident number
Add Conclusion to Incident	Incident number
Add Incident	None
Add Users	New user's name
Assign Incident to	Assignee name, incident number

Other Users	
Assign Incident to Self	Incident number
Change Incident Priority	New priority number, incident number
Change Incident Status(Open, Closed, Working)	New status, incident number(s)
Update Tracking Rule	Incident number

Dependencies: FPT_STM.1 Reliable time stamps (EXP)

SCW_EDC.1 System Data Collection

Hierarchical to: No other components

SCW_EDC.1.1 The TSF shall be able to collect event data from the following products:

a) ISS RealSecure 6.0 thru 7.0

b) Snort Intrusion Detection System 1.7 thru 2.0 (EXP)

Dependencies: No dependencies.

SCW_DRE.1 Data Reporting

Hierarchical to: No other components

SCW_DRE.1.1 The TSF shall be able to report collected event data using automatically generated reports. (EXP)

SCW_DRE.1.2 The TSF shall be capable of generating the following reports on a daily basis:

1	Incident Counts by Code
2	Incident Counts by Status
3	Top Alerts by Category
4	Top Alerts by Device IP Address
5	Top Alerts by Expert IP Address
6	Top Alerts by Source IP Subnet
7	Top Alerts by Source Ports
8	Top Alerts by Target IP Subnet
9	Top Alerts by Target Ports
10	Top Correlated Alerts by Category

11	Top Correlated Alerts by Device IP Address
12	Top Correlated Alerts by Expert IP Address
13	Top Correlated Alerts by Source IP Subnet
14	Top Correlated Alerts by Source Ports
15	Top Correlated Alerts by Target IP Subnet
16	Top Correlated Alerts by Target Ports
17	Top Correlated Alerts by Generic Alert Types
18	Top Correlated Source IP Addresses
19	Top Correlated Target IP Addresses
20	Top Generic Alert Types
21	Top Source IP Addresses
22	Top Target IP Addresses

(EXP)

SCW_DRE.1.3 The TSF shall be capable of generating the following reports on a weekly basis:

1	Incident Counts By Code
2	Incident Counts By Status
3	Top Correlated Generic Alert Types
4	Top Generic Alert Types

Dependencies: No dependencies.

(EXP)

5.5 SFRs With SOF Declarations

59 The overall Strength of Function (SOF) claim for the TOE is SOF-basic. The TOE enforces a minimum password length of eight characters for authentication.

60 Although cryptographic mechanisms are also probabilistic in nature and are often described in terms of strength, under CC rules AVA_SOF.1 is not applicable to cryptographic mechanisms. Therefore, FCS_COP and FCS_CKM do not require SOF declarations. As a result, the assessment of algorithmic strength for cryptographic functions does not form part of the evaluation.

6 TOE SUMMARY SPECIFICATION

61 This section presents an overview of the security functions implemented by the TOE and the Assurance Measures applied to ensure their correct implementation. Table 9 traces each IT security function to TOE security functional requirements.

Table 9: Security Function to TOE SFR Tracing

IT Security Function	TOE SFR	
TSF_INA	FIA_UAU.2	User Authentication Before Any Action
TSF_INA	FIA_UAU.7	Protected Authentication Feedback
TSF_INA	FIA_UID.2	User Identification before any action
TSF_FMT	FMT_MOF.1(1) FMT_MOF.1(2) FMT_MOF.1(3) FMT_MOF.1(4)	Management of Security Functions Behavior
TSF_FMT	FMT_SMF.1	Specification of Management functions
TSF_FMT	FMT_SMR.1	Security roles
TSF_UAL	SCW_UAL.1	User Action Log
TSF_EDC	SCW_EDC.1	System Data Collection
TSF_KMG	FCS_CKM.1	Cryptographic Key Generation
TSF_KMG	FCS_CKM.2	Cryptographic Key Distribution
TSF_KMG	FCS_CKM.4	Cryptographic Key destruction
TSF_CCS	FCS_COP.1	Cryptographic operation
TSF_CCS	FPT_ITT.1	Basic internal TSF data transfer protection
TSF_DRE	SCW_DRE.1	Data Reporting

6.1 TOE Security Functions

62 This section presents the security functions performed by the TOE to satisfy the identified SFRs in Section 5.1.1. Traceability to SFRs is also provided.

6.1.1 Identification and Authentication (TSF_INA)

63 TSF_INA, CyberWolf's I&A mechanism, is built on top of Tomcat's Java Database Connectivity (JDBC) Realms. TSF_INA allows CyberWolf to authenticate users, and look up the corresponding security roles, from the information found in a relational database accessed via JDBC APIs.

64 Each time that CyberWolf needs to authenticate a user, it will call the authenticate() method of this Realm implementation, passing the username and password that were specified by the user. Tomcat verifies the username exists in the database then extracts the password hash from the database for that user. Tomcat performs a comparison of the hashed value of the password with the password hash extracted from the database to authenticate the user. Following this, all of the security roles that are defined for this user are accumulated and the user is permitted access using

those roles. If the user is not authenticated, the use will be returned to the main login screen. After three bad attempts, the browser window is closed and Tomcat issues an "Unauthorized" message. The set of security roles for this user at the time of authentication are cached so that additional validation of user roles can be verified without going back to the database every time.

65 CyberWolf stores username and the MD5 hash of the user's password in a table in the external Oracle database. It is essential that the security of the Oracle database be maintained and that a unique username and password be associated with the database. It is essential that the Oracle database not allow access from the network. This implies that CyberWolf and the Oracle database run on the same computer system. While it is technically possible to run Oracle on a separate host, this configuration is not recommended and is outside the scope of the evaluation.

66 **Functional Requirements Satisfied:** FIA_UAU.2, FIA_UAU.7, FIA_UID.2

6.1.2 Security Management (TSF_FMT)

CyberWolf differentiates between 4 user roles. Each user role is assigned a limited number of security functions that the role can perform on the TOE. The administrator role has the ability to enable, disable, or modify the behavior of all security functions. The senior incident handler has the ability to assign, modify, and close incidents. The junior incident handler has the ability to assign, modify, and close incidents that are assigned to them already. The read-only user only has the ability to view incidents and reports.

67 **Functional Requirements Satisfied:** FMT_MOF.1, FMT_SMF.1, FMT_SMR.1

6.1.3 User Action Log (TSF_UAL)

68 CyberWolf collects a log of certain user actions that result in changes to the CyberWolf database. The logs include the user name performing the action, the type of event, the date and time of the event, and the outcome of the event. The log also includes any additional information that may be relevant to understanding the details of the event which occurred (e.g. the name of the user that was added).

69 The user actions which are logged include Add Action to Incident, Add Conclusion to Incident, Add Incident, Add Users, Assign Incident to Other Users, Assign Incident to Self, Change Incident Priority, Change Incident Status, and Update Tracking Rule. The user log enables the Administrator to audit user actions in order to hold individual users responsible for configuration changes made to the TOE (such as adding new users) and annotation of collected data records.

70 **Functional Requirements Satisfied:** SCW_UAL.1

6.1.4 Data Collection (TSF_EDC)

71 CyberWolf collects data from the ISS RealSecure and Snort Intrusion Detection Sensors by utilizing its RealSecure and Snort Device Experts. RealSecure versions 6.0 thru 7.0 are supported. Snort versions 1.8.X through 2.0 are supported for Linux and 1.8.1 through 2.0 are

supported for Windows. CyberWolf Device Experts consist of a common logic core, sensor(s), and knowledge about the data source's events. All Device Experts are written in Java and are compiled and run using Java VM version 1.3.x.

72 The Device Expert common logic core is the same for all CyberWolf Device Experts. It provides the framework for collecting data from almost any data source without requiring specialized logic. This is made possible through the use of Java reflection and use of a common Java interface for all Device Expert sensors.

73 The Device Expert sensors actually perform the event collection from the data source. The Device Expert common logic core calls a method named ReadDevice implemented by the sensor to access the data source and read its contents. There are several sensors available for use with the Device Expert logic core. The RealSecure Device Expert utilizes the DatabaseSensorSequence sensor. The Snort Device Expert utilizes the LogSensor sensor. The DatabaseSensorSequence sensor utilizes the Java Database Connectivity (JDBC) API to connect to the RealSecure EventCollector database and Java objects from the Java.SQL package. Each row in the database has a unique event ID column that contains an incrementing integer value maintained by the ISS RealSecure product. As the sensor reads the database it tracks the last event ID read so it "knows" where to start reading the next time the ReadDevice method is called from the Device Expert common logic core. The LogSensor utilizes Java objects from the Java.IO package to access and read events from the Snort event log. As the sensor reads the log it tracks the last file offset read so it "knows" were to start reading the next time the ReadDevice method is called from the Device Expert common logic core.

74 The Device Expert knowledge content is what distinguishes one Device Expert from another. Knowledge is applied to a data source's raw events in three major steps: Translation, Standard Alert Table Look Up and Rules. Translation is the initial step in the process and is actually performed by the sensor in use. Translation takes the raw event as reported by the data source and puts it in name value pair format using names known throughout the CyberWolf system. Translation also acts as first level filter since only events known to be related to security are translated. Most importantly, Translation identifies the raw event as a particular type of event. In the case of Snort and RealSecure, the security signatures reported by those data sources are used directly. Standard Alert Table Look Up takes the translated event and maps it to a generic classification and provides a simplified description of the event. Rules are the last step in the data collection process. Rules are used to determine if the translated event is important enough to send to the CyberWolf Manager component for correlation and/or saving to the database.

75 **Functional Requirements Satisfied:** SCW_EDC.1

6.1.5 Key Management (TSF_KMG)

76 Each CyberWolf component generates its own unique symmetric secret key of the maximum bit size supported by the encryption key algorithm chosen at install time. CyberWolf components, with the exception of the Monitor, generate a new key at random intervals between an adapted minimum and maximum interval. CyberWolf components register their current key with the CyberWolf Monitor component at startup, anytime a connection is re-established with the Monitor and each time a new key is generated. The CyberWolf Monitor maintains a list of

active keys and the associated component in memory. When a CyberWolf component needs to communicate with another CyberWolf component, it requests the desired component's connection information from the Monitor which includes the current active encryption key for the desired component. As the CyberWolf components generate a new secret key, the old key remains available for a period of 30 seconds from the time the new key is available. The idea is to ensure any messages that were in transit at the time the new encryption key went into effect (i.e., encrypted with the old key) are still able to be processed by the receiving component. The old key is destroyed by overwriting it in memory.

77 **Functional Requirements Satisfied:** FCS_CKM.1, FCS_CKM.2, FCS_CKM.4

6.1.6 Communications Security (TSF_CCS)

78 CyberWolf utilizes encryption for all message traffic between all CyberWolf components. The user chooses the encryption algorithm used at install time. The available encryption algorithms include: DES, TripleDES, and Blowfish. The implementation of the encryption algorithms is provided by Sun's Java Development group, in the Java Cryptography Extension (JCE) API version 1.2.1. When the CyberWolf Monitor is selected for install, the install wizard produces a unique symmetric secret key of the maximum length supported by the selected encryption algorithm to use as the shared secret key. That key is then encrypted with a password of a specific length, composed of pseudo random bytes using specific values for the Salt and Iteration parameters. The install wizard then writes the password and the password encrypted symmetric key to the hard drive in the install directory of the CyberWolf Monitor in the form of two files. The chosen encryption algorithm is embedded within the password encrypted symmetric key file. These two files must be manually copied to each machine hosting a CyberWolf component in order for the component to communicate with the rest of the CyberWolf Monitor component. The CyberWolf Monitor component uses the symmetric key generated by the install wizard to decrypt messages sent to it by other CyberWolf components.

79 As each component starts, it reads the shared secret information from the two files into memory, then generates its own symmetric key to use to decrypt messages sent to it. Using the symmetric secret key read from the shared secret files, each component registers its current contact information with the CyberWolf Monitor component. The contact information includes the components own symmetric secret key. As the CyberWolf Monitor component receives contact information from active CyberWolf components, it stores the contact information in memory in a hash map keyed by the registration name of the CyberWolf component.

80 In order for one CyberWolf component to contact another, it must first request the contact information for the destination component from the CyberWolf Monitor component by sending a properly formatted connection request message encrypted with the shared symmetric secret key. When the CyberWolf Monitor receives a properly formatted connection request message, it sends the requested information back to the requesting component and sends a message to the requested component containing the connection information from the requesting component. The messages from the CyberWolf Monitor are encrypted using the respective encryption keys for the components. Once this process completes, the requesting component may then

communicate with the requested component directly using the contact information without any further involvement of the CyberWolf Monitor.

81 All CyberWolf components except the CyberWolf Monitor implement encryption key rotation. The components symmetric secret key is regenerated at pseudo random intervals governed by a configurable maximum and minimum interval. These intervals are set at one hour for the minimum interval and three hours for the maximum interval by default. Once a component generates its key, the interval for the next key rotation is computed by the formula:

82 NextUpdate = ((MaxInterval - MinInterval) * (random value) + MinInterval). Where "random value" is a floating-point value between 1 and 0.

83 All CyberWolf components implement a Denial Of Service (DOS) resistance technique. Once a new connection is accepted, the source of the connection must send a properly formatted within a specified time limit. The time limit is hard coded to 1 second. If a properly formatted message doesn't arrive within one second of the connection being established, the connection is dropped and the connected socket is closed. Also, if a message arrives within the time limit but is not properly formatted, the connection is dropped as well.

84 **Functional Requirements Satisfied:** FCS_COP.1, FPT_ITT.1

6.1.7 Data Reporting (TSF_DRE)

85 TSF_DRE is an automatic reporting feature. Reporting is done in both real-time for listing alerts and incidents and generated graphically in predefined intervals. Reports are viewable by all valid CyberWolf users. Graphical reports are generated on a daily and weekly basis. By default, daily reports are generated at 2:00 AM everyday and weekly reports are generated on every Wednesday at the same time specified for the daily reports (i.e. 2:00 AM). Users must manually edit the reports configuration file if the desired reporting schedule does not match the default settings. CyberWolf reports are generated directly from the data in the database at the time the report is run. The actual charts and graphs are generated in HTML using the Java charting library MonarchCharts version 1.4.0 from Singleton Labs.

86 The daily reports include 22 specific reports and one composite report containing the results of all 22 specific reports. The following are the reports generated on a daily basis: Incident Counts By Code, Incident Counts By Status, Top Alerts By Category, Top Alerts By Device IP Address, Top Alerts By Expert IP Address, Top Alerts By Source IP SubNet, Top Alerts By Source Ports, Top Alerts By Target IP SubNet, Top Alerts By Target Ports, Top Correlated Alerts By Category, Top Correlated Alerts By Device IP Address, Top Correlated Alerts By Expert IP Address, Top Correlated Alerts By Source IP SubNet, Top Correlated Alerts By Source Ports, Top Correlated Alerts By Target IP SubNet, Top Correlated Alerts By Target Ports, Top Correlated Generic Alert Types, Top Correlated Source IP Addresses, Top Correlated Target IP Addresses, Top Generic Alert Types, Top Source IP Addresses and Top Target IP Addresses. The composite report is named daily_stats.

87 The weekly reports include four specific reports and one composite report containing the results of all four specific reports. The following are the reports generated on a weekly basis: Incident

Counts By Code, Incident Counts By Status, Top Correlated Generic Alert Types, and Top Generic Alert Types. The composite report is named weekly_stats.

88 Up to the minute, real-time data is also reported through the GUI. Incidents and Alerts are available in a browsable, web interface which provides up to the minute displays of the incidents and alerts received by the system each time the page is loaded. Incident and Alert lists can be filtered and sorted as necessary and users can add conclusions, notes, and actions to each incident.

89 **Functional Requirements Satisfied:** SCW_DRE.1

6.2 Assurance Measures

90 The TOE satisfies CC EAL2 assurance requirements. Table 10 identifies the Configuration Management, Delivery and Operation, Development, Guidance Documents, Testing, and Vulnerability Assessment Assurance Measures applied by Symantec to satisfy the CC EAL2 assurance requirements.

Table 10: EAL2 Assurance Requirements

Assurance Component	How requirement will be met
ACM_CAP.2 Configuration Items	The vendor provided configuration management documents and a Configuration Item list.
ADO_DEL.1 Delivery Procedures	The vendor provided delivery procedures.
ADO_IGS.1 Installation, Generation and Startup procedures	The vendor provided secure installation, generation and start up procedures.
ADV_FSP.1 Informal function specification	The vendor provided an informal function specification.
ADV_HLD.1 Descriptive high-level design	The vendor provided a descriptive high-level design document.
ADV_RCR.1 Informal correspondence demonstration	The informal correspondence demonstration is provided in the design documentation. ST to FSP in the FSP, FSP to HLD in the HLD.
AGD_ADM.1 Administrator Guidance	The vendor submitted a system administration manual.
AGD_USR.1 User Guidance	The vendor submitted a user guide.
ATE_COV.1 Evidence of coverage	The analysis of test coverage was submitted in the evaluation evidence.
ATE_FUN.1 Functional testing	The test evidence was submitted to the CCTL.

Assurance Component	How requirement will be met
ATE_IND.2 Independent testing - sample	The laboratory used development evidence submitted by the vendor along with functional testing evidence as a baseline for an independent test plan.
AVA_SOF.1 Strength of Function	The vendor submitted an analysis of the SOF for the password.
AVA_VLA.1 Independent vulnerability analysis	The vendor submitted vulnerability analysis was confirmed. The laboratory conducted an independent vulnerability assessment by building on the vendor's. The laboratory conducted penetration testing.

7 PROTECTION PROFILE (PP) CLAIMS

91 The TOE does not claim conformance to a PP.

8 RATIONALE

92 This section demonstrates the completeness and consistency of this ST by providing justification for the following:

Traceability — The security objectives for the TOE and its environment are explained in terms of threats countered and assumptions met. The SFRs are explained in terms of objectives met by the requirement. The traceability is illustrated through matrices that map the following:

- security objectives to threats encountered
- environmental objectives to assumptions met
- SFRs to objectives met

Assurance Level — A justification is provided for selecting an EAL2 level of assurance for this ST.

SOF — A rationale is provided for the SOF level chosen for this ST.

Dependencies — A mapping is provided as evidence that all dependencies are met.

8.1 Security Objectives Rationale

93 This section demonstrates that all security objectives for the TOE are traced back to aspects of the identified threats to be countered and/or aspects of the organizational security policies to be met by the TOE. The security objective rationale for the TOE and the environment are summarized in Tables 11 and 12.

Table 11: Security Objectives Rationale

Objective	Threat Organizational Security Policy Assumption	Rationale
O.COLLECT	P.DATA_COLLECT	O.COLLECT supports P.DATA_COLLECT by requiring the TOE to collect event data from supported security products. This is accomplished through use of the Device Experts subsystem.
O.REPORT	P.REPORT	O.REPORT satisfies P.REPORT since CyberWolf generates reports and notifications.

Objective	Threat Organizational Security Policy Assumption	Rationale
O.ADMIN	P.ADMIN	O.ADMIN satisfies P.ADMIN by ensuring that there is a set of functions for administrators to use. Management functions of the TOE shall be restricted to the Administrator.
O.SEP_ROLE	P.ADMIN T.ALTER_CONFIG	P.ADMIN requires management functions be restricted to the administrator. O.SEP_ROLE maintains that the administrator is a separate, defined role. O.SEP_ROLE mitigates the risk of anyone but authorized Administrators altering the configuration, satisfying T.ALTER_CONFIG.
O.CONF_DATA	T.ALTER_CONFIG	Maintaining confidential data transactions between TOE components, O.CONF_DATA, mitigates the risk of unauthorized users sending instructions to TOE components which would alter their configuration, T.ALTER_CONFIG.
O.LOGGING	T.ALTER_CONFIG P.ACCACT	Logging user actions, O.LOGGING, will produce an audit log which will allow Administrators to indicate/identify any possible attempts to access/modify the security policy configured on the device, countering T.ALTER_CONFIG. Logging will also hold human users of the TOE accountable for their actions, P.ACCACT, in the event that a user weakens or violates the security policy of the TOE by changing the configuration.

Table 12: Security Objectives Rationale for the Environment

Objective	Threat Organizational Security Policy Assumption	Rational
OE.INSTALL	A.INSTALL A.MANAGE A.NO_EVIL_ADM A.PROCEDURE	OE.INSTALL is met by A.INSTALL, A.MANAGE, A.NO_EVIL_ADM, and A. PROCEDURE. These environmental assumptions specify the need for the TOE hardware and software to be delivered, installed

Objective	Threat Organizational Security Policy Assumption	Rational
		and setup in accordance with documented delivery and installation/setup procedures. Additionally, they call for one or more competent Authorized Administrators, which are not willfully negligent, nor hostile, and will follow and abide by the Administrator documentation, assigned to manage the TOE and the security functions it performs.
OE.PHYSICAL	A.PHYSICAL_PROTECT	OE.PHYSICAL is met by the A.PHYSICAL_PROTECT environmental assumption. This assumption acknowledges the need for the TOE to be located within facilities providing controlled access to prevent unauthorized physical access.
OE.DAC	A.INSTALL, A.MANAGE, A.NO_EVIL_ADM, A.PROCEDURE, A.CHANGE_PASSWORD A.ACCESS_CONTROL	A.INSTALL, A.MANAGE, A.NO_EVIL_ADM, and A.PROCEDURE ensure that the system Administrators have provided and will maintain an Operating System environment for the TOE which includes discretionary access control. This calls for one or more competent Authorized Administrators, which are not willfully negligent, nor hostile, and will follow and abide by the Administrator documentation, to maintain this DAC. A.CHANGE_PASSWORD maintains the security of the TOE by protecting the password from long term attack. A.ACCESS_CONTROL ensures that discretionary access control exists in the operating system the TOE is installed on top of.
OE.PLATFORM_SUPPORT	A.INSTALL, A.MANAGE, A.RELIABLE_TIME	A.INSTALL ensures that the TOE was installed to a platform which fully supports it. A.MANAGE ensures that TOE is maintained by competent administrators, who are not willfully negligent, who will ensure that the platform does not fail to support the TOE overtime because of incorrect hardware or software operation which may develop. A.RELIABLE_TIME ensures that the host operating system provides reliable time to the TOE.

8.2 Security Requirements Rationale

94 This section provides evidence that demonstrates that the security objectives for the TOE and the IT environment are satisfied by the security requirements.

95 These mappings demonstrate that all TOE security requirements can be traced back to one or more TOE security objective(s), and all TOE security objectives are supported by at least one security requirement.

8.2.1 Rationale For TOE Security Requirements

96 This section provides evidence demonstrating that the security objectives of the TOE are satisfied by the security requirements. Table 13 provides the security requirement to security objective mapping and a rationale to justify the mapping. Table 14 maps TOE SFRs to specific security objectives.

Table 13: Security Requirement to Objective Mapping

SFR	Rationale
FCS.CKM.1	Ensures that keys used for confidential communication between TOE components are generated. This SFR traces back to and aids in meeting the following objective(s): O.CONF_DATA
FCS.CKM.2	Ensures that keys used for confidential communications between TOE components are distributed in accordance with the TOE's key distribution policy. This SFR traces back to and aids in meeting the following objective(s): O.CONF_DATA
FCS.CKM.4	Ensures that keys used for confidential communications between TOE components are destroyed by overwriting when no longer needed so that all past communications remain confidential. This SFR traces back to and aids in meeting the following objective(s): O.CONF_DATA
FCS.COP.1	Ensures that data encryption is performed by the TOE to maintain the confidentiality of communications between TOE components. This SFR traces back to and aids in meeting the following objective(s): O.CONF_DATA
FIA_UAU.2	Ensures that users must be authenticated before any action. This makes administrator role management functions only available to a user who possesses the administrator password. This SFR traces back to and aids in meeting the following objective(s): O.ADMIN, O.SEP_ROLE

SFR	Rationale
FIA_UAU.7	Ensures that the user receives no useful feedback from the TOE until authenticated. This maintains that only authenticated administrators have access to TOE security policy data. This SFR traces back to and aids in meeting the following objective(s): O.ADMIN
FIA_UID.2	Ensures that all users must be identified before any action. This ensures that the administrator role is protected by a login mechanism. This SFR traces back to and aids in meeting the following objective(s): O.ADMIN, O.SEP_ROLE
FMT_MOF.1 (1)(2)(3)(4)	This SFR ensures that the TOE associates distinct security management functions with specific user roles. This SFR traces back to and aids in meeting the following objective(s): O.SEP_ROLE
FMT_SMF.1	This SFR defines that security management functions that the TOE is capable of performing and is a necessary dependency of FMT_MOF.1. This SFR traces back to and aids in meeting the following objective(s): O.ADMIN
FMT_SMR.1	This ensures that the TOE maintains separate roles for various management functions. This SFR traces back to and aids in meeting the following objective(s): O.SEP_ROLE
FPT_ITT.1	This ensures that data and instructions transferred between TOE components remains confidential from users and processes outside the TOE. This SFR traces back to and aids in meeting the following objective(s): O.CONF_DATA
SCW_UAL.1	This ensures that a security-relevant subset of user actions is logged. This SFR traces back to and aids in meeting the following objective(s): O.LOGGING
SCW_EDC.1	This ensures that event data is collected from the various systems the TOE's device experts are installed on. This SFR traces back to and aids in meeting the following objective(s): O.COLLECT

SFR	Rationale
SCW_DRE.1	This ensures that data collected by the TOE is reported in a collection of specified reports. This SFR traces back to and aids in meeting the following objective(s): O.REPORT

Table 14: TOE SFR Mapping to Objectives

	O.COLLECT	O.REPORT	O.CONF_DATA	O.LOGGING	O.ADMIN	O.SEP_ROLE
FCS_CKM.1			**X**			
FCS_CKM.2			**X**			
FCS_CKM.4			**X**			
FCS_COP.1			**X**			
FIA_UAU.2					**X**	**X**
FIA_UAU.7					**X**	
FIA_UID.2					**X**	**X**
FMT_MOF.1						**X**
FMT_SMR.1						**X**
FPT_ITT.1			**X**			
SCW_UAL.1				**X**		
SCW_EDC.1	**X**					
SCW_DRE.1		**X**				

8.3 Rationale For Assurance Level

97 This ST has been developed for automated incident response system in a secure environment. The TOE will be exposed to a low risk environment because the TOE sits in protected space where it is under almost constant supervision. Agents cannot physically access the TOE and have no means of tampering with the TOE. As such, the Evaluation Assurance Level 2 is appropriate.

8.4 Rationale For TOE Summary Specification

98 This section demonstrates that the TSFs and Assurance Measures meet the SFRs.

99 The specified TSFs work together to satisfy the TOE SFRs. Table 15 provides a mapping of SFRs to the TSFs to show that each SFR is captured within a security function.

Table 15: Mapping of SFRs to Security Functions

SFR	Name	TSF	Name
FIA_UAU.2	User Authentication Before Any Action	TSF_INA	Identification and Authentication
FIA_UAU.7	Protected Authentication Feedback	TSF_INA	Identification and Authentication
FIA_UID.2	User Identification before any action	TSF_INA	Identification and Authentication
FMT_MOF.1	Management of Security Functions Behavior	TSF_FMT	Identification and Authentication
FMT_SMF.1	Specification of Management functions	TSF_FMT	Identification and Authentication
FMT_SMR.1	Security roles	TSF_FMT	Identification and Authentication
SCW_UAL.1	User Action Log	TSF_UAL	User Action Log
SCW_EDC.1	System Data Collection	TSF_EDC	Data Collection
FCS_CKM.1	Cryptographic Key Generation	TSF_KMG	Key Management
FCS_CKM.2	Cryptographic Key Distribution	TSF_KMG	Key Management
FCS_CKM.4	Cryptographic Key destruction	TSF_KMG	Key Management
FCS_COP.1	Cryptographic operation	TSF_CCS	Communications Security
FPT_ITT.1	Basic internal TSF data transfer protection	TSF_CCS	Communications Security
SCW_DRE.1	Data Reporting	TSF_DRE	Data Reporting

8.4.1 TOE Assurance Requirements

100 Section 6.2 of this document identifies the Assurance Measures implemented by Symantec to satisfy the assurance requirements of EAL2 as delineated in the table in Annex B of the CC, Part 3. Table 16 maps the Assurance Requirements with the Assurance Measures as stated in Section 5.2.

Table 16: Assurance Measure Compliance Matrix

Assurance Measure	Configuration Management	Delivery and Operation	Development	Guidance	Test	Vulnerability Assessment
ACM_CAP.2	X					
ADO_DEL.1		X				
ADO_IGS.1		X				
ADV_FSP.1			X			
ADV_HLD.1			X			
ADV_RCR.1			X			
AGD_ADM.1				X		
AGD_USR.1				X		
ATE_COV.1					X	
ATE_FUN.1					X	
ATE_IND.2					X	
AVA_SOF.1						X
AVA_VLA.1						X

8.4.2 TOE SOF Claims

101 The overall TOE SOF claim is SOF-basic because this SOF is sufficient to resist the threats identified in Section 3.2. Section 8.1 provides evidence that demonstrates that TOE threats are countered by the TOE security objectives. Sections 8.2.1 and 8.2.2 demonstrate that the security objectives for the TOE and the TOE environment are satisfied by the security requirements. The SOF-basic claim for the TOE applies because the TOE protects against an unskilled attacker with no special tools from accessing the TOE.

8.5 Rationale For SFR and SAR Dependencies

102 Table 17 is a cross-reference of the functional components, their related dependencies, and whether the dependency was satisfied.

Table 17: SFR Dependencies Status

Functional Component ID	Functional Component Name	Dependency (ies)	Satisfied
FCS_CKM.1	Cryptographic key generation	[FCS_CKM.2 or FCS_COP.1] FCS_CKM.4 FMT_MSA.2	No. With explanation

Functional Component ID	Functional Component Name	Dependency (ies)	Satisfied
FCS_CKM.2	Cryptographic key distribution	[FDP_ITC.1 or FCS_COP.1] FCS_CKM.4 FMT_MSA.2	No. With explanation
FCS_CKM.4	Cryptographic key destruction	[FDP_ITC.1 or FCS_COP.1] FMT_MSA.2	No. With explanation
FCS_COP.1	Cryptographic operation	[FDP_ITC.1 or FCS_COP.1] FCS_CKM.4 FMT_MSA.2	No. With explanation
FIA_UAU.2	User Authentication before any Action	FIA_UID.1	No. With explanation
FIA_UAU.7	Protected authentication feedback	FIA_UAU.1	Yes
FIA_UID.2	User identification before any action	None	N/A
FMT_MOF.1	Management of security functions behavior	FMT_SMR.1 FMT_SMF.1	Yes
FMT_SMF.1	Specification of management functions	None	N/A
FMT_SMR.1	Security roles	FIA_UID.1	Yes
FPT_ITT.1	Basic internal TSF data protection	None	N/A
SCW_UAL.1	User Action Log	FPT_STM.1	No. With explanation.
SCW_EDC.1	External Data Collection	None	N/A
SCW_DRE.1	Data Reporting	None	N/A

103 All of the dependencies have been satisfied with the exception of:

- FMT_MSA.2
- FIA_UID.1
- FPT_STM.1

104 FIA_UID.1 was not met for FIA_UAU.2 because FIA_UID.2 was met, which is hierarchical to FIA_UID.1.

105 FMT_MSA.2 was not met for FCS_CKM.1,2,4 and FCS_COP.1 since the TOE does not perform checking for secure security attributes for its cryptographic key exchange process. This is because there are no user roles which input security attributes (the cryptographic key), instead the keys are generated automatically by each component. The cryptographic key for the Monitor is outputted to the operating system of the TOE and is copied manually by the system administrator to other components of the TOE. This key is protected by

A.ACCESS_CONTROL, fulfilling the need for discretionary access control for the TOE operating system, which only allows these keys to be read, copied, or deleted by the authorized administrator.

106 FPT_STM.1 was not met because reliable time stamps are assumed to be provided by the environment of the host machines running the TOE software. A.RELIABLE_TIME ensures that the time and date passed to the TOE by the host operating system is reliable.

107 SAR dependencies identified in the CC have been met by this ST as shown in Table 18.

Table 18: EAL2 SAR Dependencies Satisfied

Assurance Component ID	Assurance Component Name	Dependencies	Satisfied
ACM_CAP.2	Configuration items	None	NA
ADO_DEL.1	Delivery procedures	None	NA
ADO_IGS.1	Installation, generation, and start-up procedures	AGD_ADM.1	YES
ADV_FSP.1	Informal functional specification	ADV_RCR.1	YES
ADV_HLD.1	Descriptive high-level design	ADV_FSP.1, ADV_RCR.1	YES
ADV_RCR.1	Informal correspondence demonstration	None	YES
AGD_ADM.1	Administrator guidance	ADV_FSP.1	YES
AGD_USR.1	User guidance	ADV_FSP.1	YES
ATE_COV.1	Evidence of coverage	ADV_FSP.1, ATE_FUN.1	YES
ATE_FUN.1	Functional testing	None	NA
ATE_IND.2	Independent testing-sample	ADV_FSP.1, AGD_ADM.1, AGD_USR.1, ATE_FUN.1	YES
AVA_SOF.1	Strength of TOE security function evaluation	ADV_FSP.1, ADV_HLD.1	YES
AVA_VLA.1	Developer vulnerability analysis	ADV_FSP.1, ATE_HLD.1 AGD_ADM.1, AGD_USR.1	YES

8.6 Rationale for Explicitly Stated Requirements

108 SCW_UAL.1: User Action Log – was added because CC permitted operations on the TSF FAU_GEN.1 can not accurately reflect the implementation of the audit functionality of the TOE.

109 SCW_EDC.1: System Data Collection – was added to ensure coverage of the fact that the TOE will be able to collect incident data from the security products Snort and ISS RealSecure.

110 SCW_DRE.1: Data Reporting – was added to ensure coverage of the fact that the TOE will be able to report the data it collects and processes, which is the motivation behind using CyberWolf as a security tool.

8.7 Rationale for Explicitly Stated Requirements Dependencies

111 SCW_UAL.1: User Action Log – has a dependency of FPT_STM.1, reliable time stamps, so that audit logs provide reliable time stamps for each record. This dependency is not satisfied by the TOE because the underlying operating system will provide the correct and reliable time. The environmental assumption A.MANAGE ensures that the system administrator, among his duties in managing and maintaining the security of the TOE, will keep the operating system time correct.

8.8 Internal Consistency and Mutually Supportive Rationale

112 The set of security requirements provided in this ST form a mutually supportive and internally consistent whole for the following reasons:

a) The choice of security requirements is justified as shown in Sections 8.3 and 8.4. The choice of SFRs and SARs is based on the assumptions about the objectives for, and the threats to, the TOE and the security environment. This ST provides evidence that the security objectives counter threats to the TOE, and that physical, personnel, and procedural assumptions are satisfied by security objectives for the TOE environment.

b) The security functions of the TOE satisfy the SFRs as shown in Table 15. All SFR and SAR dependencies have been satisfied or rationalized as shown in Table 17 and Table 18 and described in Section 8.6.

c) The SARs are appropriate for the assurance level of EAL2 and are satisfied by the TOE as shown in Table 16. EAL2 was chosen to provide a basic level of independently assured security with the assumption that products used in these environments will meet the security needs of the environment.

d) The SFRs and SARs presented in Section 5 and justified in Sections 8.3 and 8.4 are internally consistent. There is no conflict between security functions, as described in Section 2 and Section 6, and the SARs to prevent satisfaction of all SFRs.