TechGuard Security
PoliWall-CCF v. 2.01.01

Security Target

Version 0.6
January 26, 2011

Prepared for:
TechGuard Security
743 Spirit 40 Park Drive - Suite 206
Chesterfield, MO 63005

Prepared by:
Booz Allen Hamilton
Common Criteria Testing Laboratory
900 Elkridge Landing Road, Suite 100
Linthicum, MD 21090-2950
and
TechGuard Security
1001 Boardwalk Springs Place, Suite 30
O'Fallon, MO 63368

Table of Contents

List of Figures

List of Tables

1 Security Target Introduction

This chapter presents the Security Target (ST) identification information and an overview. An ST contains the Information Technology (IT) security requirements of an identified Target of Evaluation (TOE) and specifies the functional and assurance security measures offered by the TOE. This Security Target covers the PoliWall-CCF model, version 2.01.01.

1.1 ST Reference

This section provides information needed to identify and control this ST and its Target of Evaluation. This ST targets Evaluation Assurance Level 4.

1.1.1 ST Identification

ST Title: TechGuard Security PoliWall Security Target
ST Version: 0.6
ST Publication Date: January 26, 2011
ST Author: Booz Allen Hamilton and TechGuard Security

1.1.2 Document Organization

Table 1-1: ST Organization outlines the chapters and sections of the TechGuard Security PoliWall ST. This table is to be used by the reader as a quick reference guide for chapter descriptions and document navigation. The *Chapter* column identifies the chapter name, where as the *Section* column lists the sections within the chapter. Finally, the *Description* column provides a brief description of the topics covered in each respective *Chapter*.

Chapter	Section	Description
1. ST Introduction	Security Target, TOE, and CC Identification Security Target Organization Conformance Claims Conventions, Terminology, and Acronyms Security Target Overview	Provides introductory and identifying information for the TechGuard PoliWall ST.
2. Conformance Claims	CC version CC claims PP claims Package claims	Provides an overview of the claims against which the TOE is being made for the evaluation.

Chapter	Section	Description
3. Security Problem Definition	Threats Organizational Security Policies Assumptions Security Objectives	Provides the security environment description in terms of Assumptions, Threats, Objectives (both for the TOE and the Operational Environment), and Operational Security Policies.
4. Extended Security Functional Requirements	Extended SFRs for the TOE Extended SFRs for the Operational Environment	Identifies the extended security requirements for the TOE and Operational Environment.
5. Extended Security Assurance Requirements	N/A	Identifies the extended security requirements for the evaluation.
6. Security Functional Requirements	N/A	Provides the TOE security functional requirements that will be subject to evaluation.
7. Security Assurance Requirements	N/A	Identifies the security assurance requirements that will be used to perform the development and evaluation for the TOE work products.
8. TOE Summary Specification	Physical Boundary Logical Boundary	Provides a description of the scope of the evaluation for the TOE. Also describes the functions provided by the TOE to satisfy the security functional requirements.

Chapter	Section	Description
9. TOE Summary Specification Rationale	N/A	Provides a summary mapping between the Security Functional Requirements for the TOE and the TOE's capabilities as described in the TOE Summary Specification.
10. Security Problem Definition Rationale	Security Objectives Rationale EAL4 Justification Requirement Dependency Rationale Security Functional Requirements Rationale	Provides a rationale for the chosen EAL, any deviations from CC Part 2 with regards to SFR dependencies, and a mapping of threats to assumptions, objectives, and SFRs.
11. Assurance Measures	N/A	Identifies the items used to satisfy the Security Assurance Requirements for the evaluation.

Table 1-1: ST Organization

1.1.3 **Terminology**

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

Terminology	Definition
Alarm	A message that is provided to all PoliWall administrators when a condition such as log filling up or excessive invalid logins is reached.
Alert	A SNMP Trap that is sent out when a Country or group of Countries has exceeded the trigger threshold for a Rule Group.
Command Log	System commands executed by PoliWall administrators.
Country Statistics	Tracks the number of allowed and denied packets that are processed by the PoliWall
Default Rule Groups	Serve as generic filtering targets for all ingress or egress network traffic.
Exception Lists	A list of IPv4 or IPv6 addresses or networks that the Administrator will prepare on the PoliWall. An Exception List may be used to allow or deny traffic.
IPv4 Packet Log	Data for all dropped IPv4 packets by source IP, destination IP, protocol, cause and country
IPv6 Packet Log	Data for all dropped IPv6 packets by source IP, destination IP, protocol, cause and country
Overrides	Additional country-blocking restrictions applied to a specific rule group. These countries will continue to e blocked on the resource group/interface even if the Policy for that rule group is changed to allow traffic for that country.
Policy	A grouping of a Category (Country) Map, PCELs, and Exception Lists that identify which external IP addresses are to be allowed and which are to be denied. When a Policy is bound to a Rule Group, the it is applied to all rules for the Rule Group.
PreCompiled Exception List (PCEL)	A list of IPv4 and/or IPv6 addresses that is prepared off of the TOE and then uploaded to the TOE. A PCEL may be used to allow (whitelist) or deny (blacklist) traffic. PCELs may contain up to 20 million unique IP addresses.
Pre-Shared Key	An agreed upon that secret that is used to authenticate both ends of a connection.
Remote Management Console	The user GUI that is accessed to manage the PoliWall. This is a web site that runs on the PoliWall which the administrators access via an HTTPS connection.
Remote Management Console Server	A separately purchased product used for management of multiple PoliWalls. This product is excluded from evaluation,

	but the interface between itself and the PoliWall is included. This product allows for administrators to identify configuration changes, and then select which PoliWalls should perform those changes.
Rule Groups	Identify collections of internal network resources that are to be protected. For ingress rule groups, these network resources will be services that are being offered to the outside world. For egress rule groups, these network resources will be computers that are connecting out to the outside world.
System Log	System information, warning and error messages
VPN Destination Network	The IP address (or range) of the actual network to which a VPN connection is made through the Peer Address.
VPN Peer Address	IP address of the VPN endpoint

Table 1-2: Terminology Definitions

1.1.4 Acronyms

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

Acronym	Definition
ARP	Address Resolution Protocol
CC	Common Criteria
DB	Database
HIPPIE	High-Speed Internet Protocol Packet Inspection Engine
HTTPS	Hypertext Transfer Protocol over Secure Socket Layer
IPSec	Internet Protocol Security
IT	Information Technology
MTU	Maximum Transmission Unit
NTP	Network Time Protocol
OS	Operating System
PCEL	PreCompiled Exception List
PEM	Privacy Enhanced Mail
PSK	Pre-shared Key
RMC	Remote Management Console
SNMP	Simple Network Management Protocol
ST	Security Target
TOE	Target of Evaluation
TSF	TOE Security Function
VLAN	Virtual Local Area Network
XML	Extensible Markup Language

Table 1-3: Acronym Definitions

1.1.5 References

[1] TechGuard Security PoliWall-CCF User Guide rev 2.01.01
[2] TechGuard Security PoliWall-CCF Quick Start Guide 2.01.01

1.2 TOE Reference

TechGuard Security PoliWall-CCF ® 2.01.01

1.2.1 TOE Overview

This Security Target (ST) defines the Information Technology (IT) security requirements for the TechGuard Security PoliWall. TechGuard Security PoliWall is a network boundary device that rapidly determines the country of origin (category) for all incoming packets using HIPPIE™ (High-speed Internet Protocol Packet Inspection Engine) technology. Packets are filtered according to customer-defined policies, PCELs, and exception lists that are bound to rule groups for specific network addresses and protocols. PoliWall also provides Administrators with the ability to create maps by specifying one or more countries that should be allowed and customize their workspace via a graphical user interface.

PoliWall is initially set to a deny-all condition by default. The IP address of the Administrative Interface is 192.168.1.1, with a netmask of 255.255.255.0.

PoliWall protects networks by utilizing HIPPIE filters. Filtering of traffic is applied in several stages:

1. The source IP of the packet is compared to the REACT lists. If the source IP is found on either the REACT Auto-block list or the REACT Manually entered block list, the packet is dropped and no further processing is occurred.
2. The destination of the incoming packet/source of an outgoing packet (untrusted IP address) is examined to determine if the packet belongs to a resource group. If it does belong to a resource group, the filtering rules for that resource group are used. If not, the filtering rules for the default ingress or egress resource group are used.
3. The category code which the untrusted IP address belongs to is identified. The flow control policy at the category code level is checked against the resource group/untrusted IP address to determine if the traffic should be allowed or denied.
4. Depending on the decision at the category code level, the resource group/untrusted IP address are checked against either or both allow or deny pre-compiled exception lists (PCELS) to determine if the flow should be allowed or denied at the PCEL level.
5. Depending on the decision at the PCEL level, the resource group/untrusted IP address are checked against administrator defined allow and/or deny exception lists, this is to determine if the state of the packet should be allowed or denied at the exceptions level.
6. The final decision at this point is used, and the traffic is either allowed or dropped based on the flow control policy.

7. The TOE will also determine if the traffic should be allowed or dropped based on the bandwidth thresholds and the packet's priority which encompass the quality-of-service policy. If a packet is below the thresholds then its flow will be allowed, otherwise the packet will be dropped.

The TOE:

- Protects networks by utilizing HIPPIE country/IP address maps and applying filters to the network's traffic
- Is an administrative-based appliance that allows for four distinct roles: Security Administrator, Audit Administrator, Cryptographic Administrator and Read-Only.
- Provides administrators the ability to create filtering policies by specifying one or more countries that should be allowed
- Allows Administrators to specify additional allow/deny rules for IP networks or addresses with as much granularity as desired across the entire IP address space
- Allows Administrators to specify large allow/deny lists (PCELs) that can contain up to 20 million unique IP addresses. These PCELs are created outside of the TOE and then manually updated onto the TOE. The TOE can then receive updates to these PCELs from the Auto-Update Server.

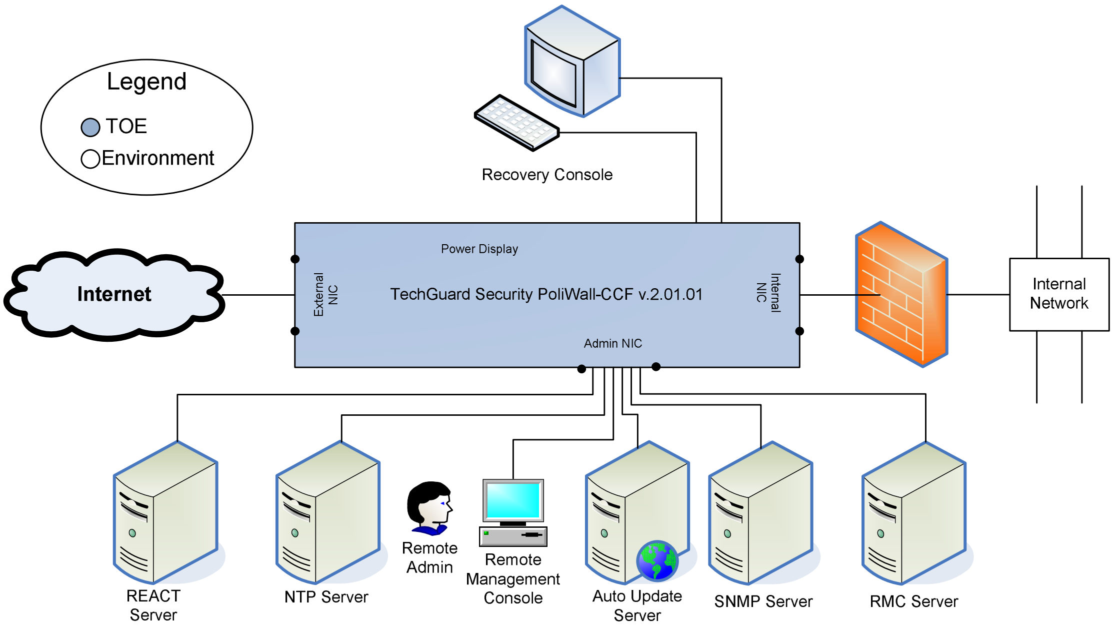

Figure 1-1: TOE Boundary

In the evaluated configuration, There are 4 physical interfaces on the PoliWall.

1. Internal (Transparent Bridging on this interface) Connects to the next appliance in the network security chain (e.g. firewall)

2. External (Transparent Bridging on this interface) Connects to the Internet (border router)
3. Administration/logging - Connects to the administrative network for administration purposes. The default IP is 192.168.1.1 and is the only NIC with an IP address.
4. Unused Port(s) (no connection)

The following security features are enforced by the TOE: Security Audit, Identification and Authentication, Security Management, User Data Protection, Cryptographic Support, Resource Utilization, Protection of the TSF, Trusted Path and TOE Access. For an explanation of each of these security classes, see section 1.3.4 Logical Boundary.

1.3 TOE Description

1.3.1 Physical Boundary

The following are the specifications for the TechGuard PoliWall-CCF 10 Gigabit hardware:

- Processor: 2x Intel Xeon E5620 @ 2.4 GHz
- Memory: 48 GB standard
- Storage: 8x Internal 2.5" HDD 300 GB
- Cryptographic Protocols: Supports, AES 256, RSA 2048, SHA1, SHA256
- System Control and Indicator Power: LED x1, HDD LED x2 on each HDD, Power on/off switch x1, LED x2 on each RJ-45 receptacle
- Number of device interfaces: 2 CX4 ports, 4 Ethernet ports (1 used, 3 unused)
- Ethernet 1, 2: 10GbE with CX4 connector or Short-Range Fiber connector
- Ethernet 3, 4, 5, 6: 10/100/1000 (GbE) with RJ-45 connector
- System Console Port: COM port x 2 (1 x Rear), RS-232 & DB-9 receptacles, USB 2.0 x 4 (2 x Rear)
- Power Supply: 2x 870 W hot swap power supply

The following are the specifications for the TechGuard PoliWall-CCF 1 Gigabit hardware:

- Processor: Intel Xeon X3430 @ 2.4 GHz
- Memory: 16 GB standard
- Storage: Internal 3.5" HDD 160 GB
- Cryptographic Protocols: Supports, AES 256, RSA 2048, SHA1, SHA256
- System Control and Indicator Power: LED x1, HDD LED, Power on/off switch x1, LED x2 on each RJ-45 receptacle
- Number of device interfaces: 4 Ethernet ports (3 used, 1 unused)
- Ethernet 1, 2: 10/100/1000 (GbE) with RJ-45 connector or Short-Range Fiber connector
- Ethernet 3, 4: 10/100/1000 (GbE) with RJ-45 connector

- System Console Port: COM port x 2 (1 x Rear), RS-232 & DB-9 receptacles, USB 2.0 x 4 (2 x Rear)
- Power Supply: 250 W power supply

The following are the specifications for the TechGuard PoliWall-CCF 10 Megabit and 50 Megabit hardware:

- Processor: Intel Atom D510 @ 1.66 GHz
- Memory: 4 GB standard
- Storage: Internal 2.5" HDD 160 GB
- Cryptographic Protocols: Supports, AES 256, RSA 2048, SHA1, SHA256
- System Control and Indicator Power: LED x1, HDD LED x2, Power on/off switch x1, LED x2 on each RJ-45 receptacle
- Number of device interfaces: 4 Ethernet ports (3 used, 1 unused)
- Ethernet 1, 2, 3, 4: 10/100/1000 (GbE) with RJ-45 connector
- System Console Port: COM port (1 x Rear), RS-232 & DB-9 receptacles, USB 2.0 x 2 (2 x Rear), PS/2 Ports (2 x Rear)
- Power Supply: 200 W power supply

1.3.2 **TOE Components**

1.3.2.1 PoliWall

PoliWall is a network boundary device that can be rapidly deployed in-line with the network it protects, requiring no changes to an existing network. It uses HIPPIE country maps to filter packets by continent, registry, country, IP range or specific IP addresses. Unlike a traditional firewall, PoliWall is not configured in a NAT or Route mode. Instead, PoliWall is a Layer 2 bridge that filters traffic in-line. Since the device operates at Layer 2 of the OSI model, network IP addresses are not visible or searchable by anyone outside of the network, putting it out of reach of attackers. A transparent bridge reduces the configuration complexity and saves time. In addition to its use in large corporate and government networks, it is ideal for branch offices and smaller networks which may consist of a single WAN connection and a router. The bridge can be configured by an in-house IT team, and shipped to a branch location.

1.3.3 **Components in the Operational Environment**

1.3.3.1 NTP Server

The Network Time Protocol Server is used to assure accurate synchronization of computer clock times in a network of computers. It also synchronizes the PoliWall's clock with the other TOE-associated servers.

1.3.3.2 Auto Update Module

The Auto Update Module downloads the latest IP/Country Allocation information and Category Codes daily to the TOE for filtering of network traffic. This will also be used to download updates to the PCELs daily to the TOE for updates.

1.3.3.3 SNMP Server

A client may poll the TOE via the Simple Network Management Protocol (SNMP) Server to gather statistics for the traffic flowing through the TOE. Also, the TOE may be configured to send SNMP traps out to a specified external server when certain events occur, such as raising an alert to the Remote Management Console.

1.3.3.4 Remote Management Console (RMC) Server

The TOE may connect up to the Remote Management Console (RMC) Server to get configuration updates, such as new policies, resource group definitions, or exceptions. A user may log into the RMC Server and schedule changes to occur on many PoliWalls from one centralized server instead of having to log on to each PoliWall.

1.3.3.5 REACT Server

A REACT Server may connect up to the PoliWall, authenticate, and then instruct the PoliWall to automatically block traffic from specific IP addresses for a period of time. These REACT Servers may be integrated into IDS units and provide fully automated blocking capabilities. An Administrator must configure the REACT Servers before the PoliWall will respond to them.

1.3.4 Logical Boundary

The logical boundaries of the TOE are described in the terms of the security functionalities that the TOE provides to the systems that utilize this product for information flow control.

The logical boundary of the TOE will be broken down into seven security classes: Security Audit, Identification and Authentication, Security Management, User Data Protection, Cryptographic Support, Resource Utilization, Protection of the TSF, Trusted Path and TOE Access. Listed below are the security functions with a listing of the capabilities associated with them:

1.3.4.1 Security Audit

Audit Logs

Included in the TOE is a Comprehensive Logging Utility that maintains large rotating log histories indexed for quick access and handles large sets of information that are available for analysis. The TOE provides the following logs that are indexed for quick access and searching:

- **Command Logs -** System commands executed by PoliWall administrators.
- **IPv4 Packet Logs -** Data for all dropped IPv4 packets by source IP, destination IP, protocol, cause and country.
- **IPv6 Packet Logs -** Data for all dropped IPv6 packets by source IP, destination IP, protocol, cause and country.
- **Message Logs -** Shows system information, warning and error messages.

These logs are maintained on the TOE as the following: Command Log Database, IPv4 Packet Log Database, IPv6 Packet Log Database, and Message Log Database.

The TOE records the (1) date and time of the event, (2) type of event, (3) subject identity (if applicable), and the outcome of the event (success or failure) within each audit record.

The IPv4 and IPv6 addresses of external Syslog servers can be configured for each address space on a maximum of four servers. All log configurations and modifications take effect immediately and will persist when the box is rebooted if the running configuration is saved. However, the System Log Server is not included in the evaluated configuration. The TOE has the ability to associate the logs/audit data with the Administrator who initiated the audit event(s).

The following rules apply to data pertaining to or extracted from the audit trail:

- All Administrators have the ability to read data from the audit trail, with the exception of those prohibited from reading such data. That data must be presented in an interpretable fashion for the Administrator(s) viewing it.
- Searching and sorting of the audit data is permitted based on user identity and a range of one or more or both of dates and times.
- Audit log data should be protected against unauthorized deletion (the Audit Administrator is the only Administrator allowed to delete records) and/or modifications to the records contained in the audit trail (no Administrator is authorized to make modifications to audit records).
- If the audit trail has exceeded its threshold, an alert will be sent to the Security Administrator.
- If the audit trail’s threshold has been reached and is full, the oldest stored audit records will be overwritten. Once this occurs a message will be sent to the remote management console notifying of such an occurrence.

Security Alarms

The TOE is able to generate security alarms when a potential security violation occurs, thus notifying the Security Administrator of such an event. The Security Administrator will be immediately notified of this alarm during their remote session. Some of these alarms occur when there are severe events that will affect the TOE and require it to enter Maintenance Mode. These specific alarms are failure of a self-test and a log filling up. The Security Administrator may configure the PoliWall to not enter maintenance mode when logs are full and instead automatically overwrite the oldest log records. Rules will be applied by the Security Administrator on how these audited events will be monitored, which will include:

- Excessive number of authentication failures by a Administrator has resulted in an account being locked out. This alarm will never cause the PoliWall to enter Maintenance Mode.
- An audit log (IPv4 Packet Log, IPv6 Packet Log, or Message Log) has reached the warning level threshold. This will never cause the PoliWall to enter Maintenance Mode.
- An audit log (IPv4 Packet Log, IPv6 Packet Log, or Command Log) has become full. This will cause the PoliWall to enter Maintenance Mode if configured to do so by the Security Administrator.
- A Self-Test has failed. This will always cause the PoliWall to enter Maintenance Mode.
- An Automatic Update failed. This will never cause the PoliWall to enter Maintenance Mode.

1.3.4.2 Cryptographic Support

The TOE utilizes cryptography across several different areas:

- Between the TOE and web interfaces
- Auto-Updating
- IPsec
- NTP
- SNMP
- Communications with the Remote Management Console (RMC) Server
- Communications with the REACT Servers

It is essential that the TOE compensate for the generation, destruction, and encryption of keys that are produced. The following chart illustrates how each entity handles those keys:

Purpose	Usage	Algorithm	Size	Standard
Key Generation	[redacted]	RSA	2048	RFC 2313
Key Destruction	[redacted]	Key Zeroization	[redacted]	No Standard.
Crypto Operation (1)	Encryption/decryption	AES	256	RFC 3268
Crypto Operation (2)	Cryptographic Hashing	SHA-1	160	RFC 3174
Crypto Operation (3)	Cryptographic Hashing	SHA-256	256	FIPS 180-2

SHA-256 is the preferred hashing mechanism and is used whenever possible for the TOE. However some protocols supported by the TOE (SNMP and IPSEC) require SHA-1 for hashing instead of SHA-256.

OpenSSL-FIPS version 1.2 is used by the TOE. The FIPS compliance is currently vendor-asserted, rather than FIPS-asserted.

1.3.4.3 Identification & Authentication

In order to authenticate to the TOE and perform TOE processes, Administrators must either enter (1) their username and password or (2) their username, password, and client certificate which will be defined by the Security Administrator. Upon attempting to authenticate the TOE, Administrators will have anywhere between 2 and 25 attempts at successfully logging in. The amount of attempts is configuration by the Security Administrator, and when that limit is reached, the Administrator will be locked out from logging in and subsequently performing TOE operations. There are two ways that an account can be unlocked – either manually by the Security Administrator or automatically when the specified time from the account locking has elapsed. If authentication and identification has been successfully completed, the Administrator's attributes associated with the role will be displayed/granted.

Password Policy

The TOE comes preconfigured with mechanisms for creating a password and strictly enforces them. The mechanisms put in place for password creation are:

- must be an 8 character minimum
- must be at least 3 of the following 4 metrics: uppercase characters, lowercase characters, numbers, symbol
- is not one of the previous # used passwords, where # is definable by the Security Administrator
- has a maximum life of # days, where # is definable by the Security Administrator
- has a minimum life of # days, where # is definable by the Security Administrator
- has a maximum authentication attempts of # before a Administrator is locked out, where # is definable by the Security Administrator
- has a lockout duration of # minutes, where # is definable by the Security Administrator
- has a maximum inactive session of # minutes before re-authentication is required, where # is definable by the Security Administrator
- has a minimum session of # minutes before re-authentication is required, where # is definable by the Security Administrator

The only action this is permitted to be performed without authenticating to the TOE is ICMP (ping). This is wholly up to the discretion of the Security Administrator whether or not they will allow this action to be enabled or disabled without authenticating to the TOE; all other TOE actions require Administrators to properly authenticate to the TOE.

The TOE allows for the association of a Administrator's security attributes to be attributed to the Administrator acting on their behalf; the rules governing this association of attributes and the changing of those attributes will be strictly enforced by the Security Administrator.

1.3.4.4 Security Management

User/Role Association

The User/Role association information, i.e. the functions that system administrators are allowed to perform, is stored in an Object that is created for each authenticated session. The TOE tracks these sessions internally in the PoliWall process and they are associated with cookies that are set on the client.

The TOE has several roles and has the following rules associated with them:

1. Security Administrator – has the ability to perform all functions except the ability to manage cryptography and delete audit logs.
2. Audit Administrator – has the ability to delete audit records
3. Cryptographic Administrator – Manages all cryptographic functionality
4. Read-Only - has the ability to read configuration information but may not make any changes to the TOE

It is the TOE's responsibility to ensure that the following conditions are satisfied:

- all roles shall be able to access the TOE remotely; Security Administrator, Audit Administrator, and Cryptographic Administrator will be able to administer the TOE, while Read-Only will only be able to view the configuration of the TOE.
- all three Administrator roles are distinct; that is, there shall be no overlap of operations performed by each default role, with the following exceptions:
    - All roles, including Read-Only, can review the audit trail;
    - The three administrator roles can invoke the self-tests and
    - All roles, including Read-Only, can accept alarms/acknowledgements

Additionally, all administrators can disable/enable security alarms, perform self-tests, have the ability to read audit records, and can accept notifications.

The TOE can revoke and enforce rules of the security attributes associated with an Administrator's information flow policy ruleset and services available to unauthenticated Administrators.

Flow Control

The TOE enforces the Unauthenticated Information Flow Control SFP to restrict the ability to change, default, and query or modify the security attributes to the Security

Administrator. The Unauthenticated Information Flow Control SFP must also provide restrictive values for security attributes to be used to enforce the SFP (i.e. deny all network traffic). The Security Administrator is the only Administrator with the ability to specify alternative initial values to override the aforementioned default values when an object/information is being created.

Quotas

Quotas for TOE data on transport-layer connections can only be determined by the Security Administrator. If the quota has been reached, all packets above and beyond the quota will be dropped. Quotas can also be placed on controlled connection-oriented resources by the Security Administrator. If the quota has been reached for these resources, the packets will be dropped.

1.3.4.5 User Data Protection

The TOE provides for enforcement of the Unauthenticated Information Flow SFP based on:

- Source Subject
- Destination Subject
- Information
- Operations

Stateful packet inspection should occur when it is received unless associated with an established session.

The information flow will be authorized when a flow has already been established and no changes to any policies have been made. The information flow will be rejected if the request for access or services where the presumed source ID of the information received by the TOE is not included in the set of source identifiers for the source subject. Any previous information content of a resource should be made unavailable upon the allocation or reallocation of the resource from the list of objects.

1.3.4.6 Trusted Path

The TOE comes pre-installed with a self-signed SSL certificate that is used to establish a secure encrypted session to the PoliWall configuration application. The appliance includes a generic server certificate. The pre-installed certificate will be overwritten after successfully configuring and installing a new server certificate. An assurance is made that a communication channel between the TOE and another IT product that provides assured identification and protection will be established. This communication will be for the purpose of updating the system time, category code database, PCELs, connection to Remote Management Console (RMC) Server, and establishment of connections from REACT Servers.

The TOE's client CA certificate specifies the certificate authority required to issue client certificates which identify Administrators connecting to the TOE. A Certificate Revocation List may be uploaded to the TOE to prevent revoked certificates issued by the client CA certificate from establishing connections to the TOE.

The TOE will provide a trusted communications path for remote Administrators to authenticate to.

1.3.4.7 Resource Utilization

A secure, stable state must be maintained when failures to the following resources occur:

- Auto Update Daemon
- PoliWall Process
- Auditing Modules
    - Msglogd, syslogd, pktlogd, pktlog6d

In the event of the failures of the Auto Update module, PoliWall process module (remote administration functions and access control), and auditing modules (msglogd, syslogd, pktlogd, pktlog6d), the TOE will maintain and operate in a secure state until these failed subsystem have come back online. Information flow control will remain in operation during this time.

Unauthenticated data to be processed by the TOE is subjected to prioritization based on QoS and quotas. Once the data has priority, an operation is made on it based on the unauthenticated information flow control.

When the total amount of traffic reaches the configured bandwidth limit, traffic from the high QoS countries will be allowed through the PoliWall before traffic from other countries.

1.3.4.8 TOE Access

Access to the TOE is controlled by the Administrator's IP address. The TOE can terminate sessions after a given amount of time of inactivity has occurred (which is predetermined by the Security Administrator). Before a session begins, a warning will be displayed alerting the Administrator that unauthorized access to the TOE is prohibited. Denials of access to the TOE can be made according to IP address, time, and day.

1.3.4.9 Protection of the TSF

The TOE will maintain a secure state even when failures to the Auto Update, PoliWall process, msglogd, syslogd, pktlogd, and pktlog6d occur. The TOE will also maintain and provide reliable timestamps to Administrators. In order to maintain the integrity of the TOE, the TSF will run a suite of self-tests during initial start-up, periodically during normal operation, and at the request of the authorized Administrator in order to demonstrate the correct operation of the TOE. All authorized Administrators will be able to verify the integrity of TOE data and stored TOE executable code. All authorized Administrators will be able to verify the integrity of TOE data and stored TOE executable code.

1.4 Excluded from the TOE

- External System Log Server
- Updating the firmware of the TOE
- Remote Management Console Server - A separately purchased product used for management of multiple PoliWalls concurrently. This product is excluded from evaluation, but the interface between itself and the PoliWall is included. This product allows for administrators to identify configuration changes, and then select which PoliWalls should perform those changes.

1.5 TOE Type

TechGuard PoliWall provides the following: Network Boundary Device.

2 Conformance Claims

2.1 CC Version

This ST is compliant with *Common Criteria for Information Technology Security Evaluation*, CCMB-2009-07-004, Version 3.1 Revision 3 July 2009.

2.2 CC Part 2 Extended

This ST and Target of Evaluation (TOE) is Part 2 extended for EAL4 to include all applicable NIAP and International interpretations through 25 February 2009.

2.3 CC Part 3 Augmented Plus Flaw Remediation

This ST and Target of Evaluation (TOE) is Part 3 augmented plus flaw remediation for EAL4 to include all applicable NIAP and International interpretations through 25 February 2009.

2.4 PP Claims

This ST does not claim Protection Profile (PP) conformance.

2.5 Package Claims

This TOE has a package claim of EAL 4.

2.6 Package Name Conformant or Package Name Augmented

This ST and Target of Evaluation (TOE) is conformant to EAL package claims augmented with ALC_FLR.2 and ASE_TSS.2**.**

2.7 Conformance Claim Rationale

There is no Conformance Claim rationale for this ST.

3 Security Problem Definition

3.1 Threats

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 moderately sophisticated. The following are threats addressed by the TOE.

T.ADDRESS_MASQUERADE A user on one interface may masquerade as a user on another interface to circumvent the TOE policy.

T.ADMIN_ERROR An administrator user may incorrectly install or configure the TOE, or install a corrupted TOE resulting in ineffective security mechanisms.

T. ADMIN_ROGUE An administrator's intentions may become malicious resulting in user of TSF data being compromised.

T.AUDIT_COMPROMISE A malicious user or process may view audit records, cause audit records to be lost or modified, or prevent future audit records from being recorded, thus masking a User's action.

T.CRYPTO_COMPROMISE A malicious user or process may cause key, data, or executable code associated with the cryptographic functionality to be inappropriately accessed (viewed, modified, or deleted), thus compromise the cryptographic mechanism and the data protected by those mechanisms.

T.FLAWED_DESIGN Unintentional or intentional errors in requirements speciation or design of the TOE may occur leading to flaws that may be exploited by a malicious user or program.

T.FLAWED_IMPLEMENTATION Unintentional or intentional errors in implementation of the TOE design may occur, leading to flaws that may be exploited by a malicious user or program.

T.MALICIOUS_TSF_COMPROMISE A malicious user or process may cause TSF data or executable code to be inappropriately accessed (viewed, modified, or deleted).

T.MASQUERADE An unauthenticated user may masquerade as an authorized user or an authorized IT entity to gain access to data or TOE resources.

T.POOR_TEST Lack of or insufficient tests to demonstrate that all TOE security functions operate correctly (including in a fielded TOE) may result in incorrect TOE behavior being undiscovered.

T.REPLAY A user may gain inappropriate access to the TOE by replaying authentication information, or may cause the TOE to be inappropriately configured by replaying TSF data or security attributes (captured as it was transmitted during the course of legitimate use).

T.RESIDUAL_DATA A user or process may gain unauthorized access to data through reallocation of TOE resources from one user or process to another.

T.RESOURCE_EXHAUSTION A malicious process or user may block others from TOE system resources (e.g., connection state tables) via a resource exhaustion denial of service attack.

T.SPOOFING An entity may mis-represent itself as the TOE to obtain authentication data.

T.UNATTENDED_SESSION A user may gain unauthorized access to an unattended session.

T.UNAUTHORIZED_ACCESS A user may gain access to services (by sending data through or to the TOE) for which they are not authorized according to the TOE security policy.

T.UNIDENTIFIED_ACTIONS The administrator may fail to notice potential security violations, thus limiting the administrator's ability to identify and take action against a possible security breach.

T.UNKNOWN_STATE When the TOE is initially started or restarted after a failure, design flaws, or improper configurations may cause the security state of the TOE to be unknown.

3.2 Organizational Security Policies

The TOE addresses the organizational security policies described below.

P.ACCESS_BANNER The TOE shall display an initial banner describing restrictions of use, legal agreements, or any other appropriate information to which users consent by accessing the system.

P.ACCOUNTABILITY The authorized users of the TOE shall be held accountable for their actions within the TOE.

P.ADMIN_ACCESS Administrators shall be able to administer the TOE remotely through protected communications channels.

P.CRYPTOGRAPHIC_FUNCTIONS The TOE shall provide cryptographic functions for its own use, including encryption/decryption and digital signature operations.

P.VULNERABILITY_ANALYSIS_TEST The TOE must undergo appropriate independent vulnerability analysis and penetration testing to demonstrate that the TOE is resistant to an attacker possessing a medium attack potential.

3.3 Assumptions

The specific conditions listed in this section are assumed to exist in the environment in which the TOE is deployed. These assumptions are necessary as a result of practical realities in the development of the TOE security requirements and the essential environmental conditions on the use of the TOE.

3.3.1 Personnel Assumptions

None

3.3.2 Physical Assumptions

A.PHYSICAL Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the environment.

3.3.3 Logical Assumptions

A.NO_TOE_BYPASS Information cannot flow between external and internal networks located in different enclaves without passing through the TOE.

3.4 Security Objectives

3.4.1 Security Objectives for the TOE

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

O.ROBUST_ADMIN_GUIDANCE The TOE will provide administrators with the necessary information for secure delivery and management.

O.ADMIN_ROLE The TOE will provide an administrator role to isolate administrative actions.

O.AUDIT_GENERATION The TOE will provide the capability to detect and create records of security-relevant events associated with users.

O.AUDIT_PROTECTION The TOE will provide the capability to protect audit information.

O.AUDIT_REVIEW The TOE will provide the capability to selectively view audit information, and alert the administrator of identified potential security violations.

O.CHANGE_MANAGEMENT The configuration of, and all changes to, the TOE and its development evidence will be analyzed, tracked, and controlled throughout the TOE's development.

O.CORRECT_TSF_OPERATION The TOE will provide the capability to test the TSF to ensure the correct operation of the TSF in its operational environment.

O.CRYPTOGRAPHIC_FUNCTIONS The TOE shall provide cryptographic functions for its own use, including encryption/decryption, key generation and destruction and cryptographic hashing services.

O.DISPLAY_BANNER The TOE will display an advisory warning regarding use of the TOE.

O.THOROUGH_FUNCTIONAL_TESTING The TOE will provide Users with the necessary information for secure delivery and management.

O.MANAGE The TOE will provide all the functions and facilities necessary to support the administrators in their management of the security of the TOE, and restrict these functions and facilities from unauthorized use.

O.MEDIATE The TOE must mediate the flow of information between sets of TOE network interfaces or between a network interface and the TOE itself in accordance with its security policy.

O.RESIDUAL_INFORMATION The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated.

O.RESOURCE SHARING The TOE shall provide mechanisms that mitigate attempts to exhaust connection-oriented resources provided by the TOE (e.g., entries in a connection state table; Transmission Control Protocol (TCP) connections used by proxies).

O.SELF_PROTECTION The TSF will maintain a domain for its own execution that protects itself and its resources from external interference, tampering, or unauthorized disclosure.

O.SOUND_DESIGN The design of the TOE will be the result of sound design principles and techniques; the design of the TOE, as well as the design principles and techniques, are adequately and accurately documented.

O.SOUND_IMPLEMENTATION The implementation of the TOE will be an accurate instantiation of its design, and is adequately and accurately documented.

O.TIME_STAMPS The TOE shall provide reliable time stamps and the capability for the administrator to set the time used for these time stamps.

O.ROBUST_TOE_ACCESS The TOE will provide mechanisms that control a user’s logical access to the TOE and to explicitly deny access to specific users when appropriate.

O.TRUSTED_PATH The TOE will provide a means to ensure administrators are not communicating with some other entity pretending to be the TOE, and that the TOE is communicating with an authorized IT entity and not some other entity pretending to be an authorized IT entity.

O.VULNERABILITY_ANALYSIS TEST The TOE will undergo appropriate independent vulnerability analysis and penetration testing to demonstrate the design and implementation of the TOE does not allow attackers with medium attack potential to violate the TOE’s security policies.

3.4.2 Security Objectives for the operational environment of the TOE

The following security objectives for the Operational environment of the TOE must be satisfied in order for the TOE to fulfill its security objectives.

OE.CRYPTANALYTIC Cryptographic methods used in the IT environment shall be interoperable with the TOE and should be resistant to cryptanalytic attacks (i.e., will be of adequate strength to protect unclassified Mission Support, Administrative, or Mission Critical data).

OE.NO_TOE_BYPASS Information cannot flow between external and internal networks located in different enclaves without passing through the TOE.

OE.PHYSICAL Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the IT environment.

4 Extended Security Functional Requirements Definition

Security Function	Security Functional Components
Security Audit (FAU)	FAU_ARP_EXT.1 Security alarm acknowledgement

Table 4-1: Extended Security Functional Requirements for the TOE

4.1 Extended Security Functional Requirements Definition for the TOE

4.1.1.1 FAU_ARP_EXT.1 Security alarm acknowledgement

Hierarchical to: No other components.

FAU_ARP_EXT.1.1 The TSF shall take [assignment***: list of actions***]
upon the acknowledgement of a potential security violation by an administrator.

Dependencies: FAU_SAA.1 Potential violation analysis, FAU_ARP.1 Security alarms

4.2 Extended Security Functional Requirements for the Operational Environment

There are no extended Security Functional Requirements for the Operational Environment.

4.3 Proper Dependencies

All dependencies for the extended security functional requirements were derived from CC Part 2.

5 Extended Security Assurance Requirements

There are no extended Security Assurance Requirements in this ST.

6 Security Functional Requirements

6.1 Security Functional Requirements for the TOE

Security Function	Security Functional Components
Security Audit	FAU_ARP.1 Security Alarms
	FAU_ARP_EXT.1 Security Alarm Acknowledgement
	FAU_GEN.1 Audit Data Generation
	FAU_GEN.2 User Identity Association
	FAU_SAA.1 Potential Violation Analysis
	FAU_SAR.1 Audit Review
	FAU_SAR.2 Restricted Audit Review
	FAU_SAR.3 Selectable Audit Review
	FAU_STG.1 Protected Audit Trail Storage
	FAU_STG.3 Action In Case Of Possible Audit Data Loss
	FAU_STG.4 Prevention of Audit Data Loss
Cryptographic Support	FCS_CKM.1 Cryptographic Key Generation
	FCS_CKM.4 Cryptographic Key Destruction
	FCS_COP.1(1) Cryptographic Operation
	FCS_COP.1(2) Cryptographic Operation
User Data Protection	FDP_IFC.1 Subset Information Flow Control
	FDP_IFF.1 Simple Security Attributes

Security Function	Security Functional Components
	FDP_RIP.1(1) Subset Residual Information Protection
	FDP_RIP.1(2) Subset Residual Information Protection
Identification and Authentication	FIA_AFL.1 Authentication Failure Handling
	FIA_ATD.1 User Attribute Definition
	FIA_SOS.2 TSF Generation of Secrets
	FIA_UAU.1 Timing of Authentication
	FIA_UAU.5 Multiple Authentication Mechanisms
	FIA_UID.2 User Identification Before Any Action
	FIA_USB.1 User-Subject Binding
Security Management	FMT_MOF.1 Management of Security Functions Behavior
	FMT_MSA.1 Management of Security Attributes
	FMT_MSA.3 Static Attribute Initialization
	FMT_MTD.1 Management of TSF Data
	FMT_MTD.2 Management of limits on TSF Data
	FMT_REV.1 Revocation
	FMT_SMF.1 Specification of management functions
	FMT_SMR.2 Restrictions on Security Roles
Protection of the TSF	FPT_FLS.1 Failure of preservation of secure state
	FPT_STM.1 Reliable time stamps
	FPT_TST.1 TSF testing
Resource Utilization	FRU_FLT.1(1) Degraded Fault Tolerance
	FRU_FLT.1(2) Degraded Fault Tolerance

Security Function	Security Functional Components
	FRU_FLT.1(3) Degraded Fault Tolerance
	FRU_FLT.2 Limited Fault Tolerance
	FRU_PRS.1 Limited Priority of Service
	FRU_RSA.1 Maximum Quotas
TOE Access	FTA_SSL.3 TSF-Initiated Termination
	FTA_TAB.1 Default TOE Access Banners
	FTA_TSE.1 TOE Session Establishment
Trusted Path/Channels	FTP_ITC.1 Inter-TSF Trusted Channel
	FTP_TRP.1 Trusted Path

Table 6-1: Security Functional Requirements for the TOE

6.1.1 **Class FAU: Security Audit**

6.1.1.1 FAU_ARP.1 Security alarms

Hierarchical to: No other components.

FAU_ARP.1.1 The TSF shall take [assignment***: action to***

- ***immediately display of an alarm message at the remote administrator's browser, identifying the potential security violation and make accessible the audit record contents associated with the auditable event(s) that generated the alarm, at the:***
    i. ***remote administrators browser for all sessions that exist,***
    ii. ***remote administrators browser for all sessions that are initiated before the alarm has been acknowledged, and***
    iii. ***at the option of the Security Administrator, generate an audible alarm***
- ***make accessible the audit record contents associated with the auditable event(s) until it has been acknowledged***
- ***the TOE will be able to send SNMP traps for configured Alerts***

upon detection of a potential security violation.

Dependencies: FAU_SAA.1 Potential violation analysis

Application Note: The message is displayed at the remote console if an administrator is already logged in, or when an administrator logs in if the alarm message has not been acknowledged. In addition, the TOE provides an audible alarm that can be configured to sound an alarm if desired by the Security Administrator.

6.1.1.2 FAU_ARP_EXT.1 Security alarm acknowledgement

Hierarchical to: No other components.

FAU_ARP_EXT.1.1 The TSF shall take [assignment***: action to immediately display an acknowledgement message at all remote administrator's browser for all sessions that received the alarm, identifying:***

- ***a reference to the potential security violation,***
- ***a notice that it has been acknowledged,***
- ***the time of the acknowledgement, and***
- ***the user identifier that acknowledged the alarm, at the:***

upon the acknowledgement of a potential security violation by an administrator.

Dependencies: FAU_SAA.1 Potential violation analysis, FAU_ARP.1 Security alarms

6.1.1.3 FAU_GEN.1 Audit data generation

Hierarchical to: No other components.

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 [selection: **not specified**] level of audit; and
c) [assignment: ***All auditable events listed in Table 6-2 Auditable Events***].

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 (if applicable), and the outcome (success or failure) of the event; and
b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [assignment: ***information specified in column three of Table 6-2 Auditable Events below].***

Dependencies: FPT_STM.1 Reliable time stamps

Application Note: In column 3 of the table below, "if applicable" is used to designate data that should be included in the audit record if it "makes sense" in the context of the event that generates the record. For example, in FDP_IFF, packets may be allowed to *flow that do not have a transport layer component (e.g., an ICMP Echo request). For those packets, there is nothing to record with respect to the transport layer abstractions.*

Requirement	Auditable Events	Additional Audit Record Contents
FAU_ARP.1	Potential security violation was detected	Identification of what caused the generation of the alarm
FAU_ARP_EXT.1	None	The identity of the administrator that acknowledged the alarm.
FAU_GEN.1	None	
FAU_GEN.2	None	
FAU_SAA.1	Enabling and disabling of any of the analysis mechanisms	The identity of the Security Administrator performing the function
FAU_SAR.1	Opening the audit trail	The identity of the Administrator performing the function
FAU_SAR.2	Unsuccessful attempts to read information from the audit records	The identity of the administrator attempting the function
FAU_SAR.3	None	
FAU_STG.1	None	
FAU_STG.3	Actions taken due to exceeding the audit threshold	The identity of the Security Administrator performing the function
FAU_STG.4	Actions taken due to the audit storage failure	The identity of the Security Administrator performing the function
FCS_CKM.1	None	
FCS_CKM.4	None	
FCS_COP.1	None	
FDP_IFC.1	None	

FDP_IFF.1	Decisions to deny information flows	Presumed identity of source subject Identity of destination subject Transport layer protocol, if applicable Source subject service identifier, if applicable Destination subject service identifier, if applicable Identity of the inbound/outbound interface associated on which the TOE received the packet Identity of the rule that disallowed the packet flow (Country Filter, PCEL, Exception)
FDP_RIP.1	None	
FIA_AFL.1	The reaching of the threshold for the unsuccessful authentication attempts The actions (e.g. disabling of an account) taken The subsequent, if appropriate, restoration to the normal state (e.g. re-enabling of an account)	Identity of the unsuccessfully authenticated user
FIA_ATD.1	None	
FIA_SOS.2	Modifications to the password policy	Security Administrators identity
FIA_UAU.1	Successful and unsuccessful use of authentication mechanisms	Claimed identity of the user using the authentication mechanism
FIA_UAU.5	Successful and unsuccessful use	Claimed identity of the user using the authentication

	of authentication mechanisms	mechanism
FIA_UID.2	All use of the user identification mechanism used for authorized users (that is, those that authenticate to the TOE)	Claimed identity of the user using the identification mechanism
FIA_USB.1	Success and failure of binding of user security attributes to a subject	The identity of the user whose attributes are attempting to be bound
FMT_MOF.1	All modifications in the behavior of the functions in the TSF	The identity of the administrator performing the function
FMT_MSA.1	All manipulation of the security attributes	The identity of the administrator performing the function
FMT_MSA.3	None	
FMT_MTD.1	All modifications of the values of TSF data by the administrator	The identity of the administrator performing the function
FMT_MTD.2	All modifications of the limits Actions taken when the quota is exceed (include the fact that the quota was exceeded)	The identity of the administrator performing the function
FMT_REV.1	All attempts to revoke security attributes	List of security attributes that were attempted to be revoked The identity of the administrator performing the function
FMT_SMF.1	All use of the management functions	The identity of the administrator performing the function
FMT_SMR.2	Modifications to the group of users that are part of a role	User IDs that are associated with the modifications The identity of the administrator performing the function
FPT_FLS.1	None	

FPT_STM.1	Changes to the time	
FPT_TST.1	Execution of this set of TSF self tests	The identity of the administrator performing the test, if initiated by an administrator
FRU_FLT.1(1)	None	
FRU_FLT.1(2)	None	
FRU_FLT.1(3)	None	
FRU_FLT.2	None	
FRU_PRS.1	None	
FRU_RSA.1	None	
FTA_SSL.3	The termination of a remote session by the session locking mechanism	The identity of the user associated with the session that was terminated
FTA_TAB.1	None	
FTA_TSE.1	All attempts at establishment of a user session	The identity of the user attempting to establish the session For unsuccessful attempts, the reason for denial of the establishment attempt
FTP_ITC.1	All attempted uses of the trusted channel functions	Identification of the initiator and target of all trusted channels
FTP_TRP.1	All attempted uses of the trusted path functions	Identification of the claimed user identity

Table 6-2: Auditable Events

6.1.1.4 FAU_GEN.2 User identity association

Hierarchical to: No other components.

FAU_GEN.2.1 For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.

Dependencies: FAU_GEN.1 Audit data generation
FIA_UID.1 Timing of identification

Application Note: For failed login attempts no user association is required because the user is not under TSF control until after a successful identification/authentication. User in this requirement is the userid for authorized users, and a network identifier for unauthenticated network traffic.

6.1.1.5 FAU_SAA.1 Potential violation analysis

Hierarchical to: No other components.

FAU_SAA.1.1 The TSF shall be able to apply a set of rules in monitoring the audited events and based upon these rules indicate a potential violation of the enforcement of the SFRs.

FAU_SAA.1.2 The TSF shall enforce the following rules for monitoring audited events:
a) Accumulation or combination of [assignment: ***Security Administrator specified number of authentication failures, Security Administrator specified threshold for the audit trail***] known to indicate a potential security violation;
b) [assignment: ***failure to automatically update the Category Code Database, when the audit trail is full and will overwrite, any failure of the TSF self-tests***].

Dependencies: FAU_GEN.1 Audit data generation

Application Note: The intent of this requirement is that an alarm is generated (FAU_ARP.1) once the threshold for an event is met. Once the alarm has been generated it is assumed that the "count" for that event is reset to zero. The Security Administrator settable number of authentication failures in bullet a) is intended to be the same value as specified in FIA_AFL.1.1.

6.1.1.6 FAU_SAR.1 Audit review

Hierarchical to: No other components.

FAU_SAR.1.1 The TSF shall provide [assignment: ***the Administrators***] with the capability to read [assignment: ***all audit data***] from the audit records.

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

Dependencies: FAU_GEN.1 Audit data generation

6.1.1.7 FAU_SAR.2 Restricted audit review

Hierarchical to: No other components.

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.

Dependencies: FAU_SAR.1 Audit review

6.1.1.8 FAU_SAR.3 Selectable audit review

Hierarchical to: No other components.

FAU_SAR.3.1 The TSF shall provide the ability to apply [assignment: ***searches or sorting***] of audit data based on [assignment:

a) user identity
b) command type executed
c) ***ranges of one or more or both: dates and times***].

Dependencies: FAU_SAR.1 Audit review

Application Note: Audit data should be capable of being searched and sorted on all criteria specified in a +b, if applicable (i.e., not all criteria will exist in all audit records). Sorting means to arrange the audit records such that they are "grouped" together for administrative review. For example the Audit Administrator may want all the audit records for a specified source subject identity or range of source subject identities (e.g., IP source address or range of IP source addresses) presented together to facilitate their audit review. If no additional criteria are provided by the TOE to perform searches or sorting of audit data, the ST author selects "no additional criteria".

6.1.1.9 FAU_STG.1 Protected audit trail storage

Hierarchical to: No other components.

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

FAU_STG.1.2 The TSF shall be able to [selection: **prevent**] unauthorized modifications to the stored audit records in the audit trail.

Dependencies: FAU_GEN.1 Audit data generation

Application note: The only user authorized to delete the audit records is the Audit Administrator.

Application note: The TOE does not authorize the modification of the audit records to any users.

6.1.1.10 **FAU_STG.3 Action in case of possible audit data loss**

Hierarchical to: No other components.

FAU_STG.3.1 The TSF shall [assignment: ***immediately alert the administrators by displaying a message at the remote management console when an administrative session exists for each of the defined administrative roles***] if the audit trail exceeds [assignment: ***a Security Administrator settable percentage of storage capacity***].

Dependencies: FAU_STG.1 Protected audit trail storage

Application Note: As with FAU_ARP.1, the TSF displays a message at the remote console if an administrator that is already logged in, or when an administrator logs in. This requirement specifies that the message is sent to the first established session for each of the defined roles to ensure someone in the administrator staff is aware of the alert as soon as possible.

6.1.1.11 FAU_STG.4 Prevention of audit data loss

Hierarchical to: FAU_STG.3 Action in case of possible audit data loss

FAU_STG.4.1 The TSF shall [selection: **overwrite the oldest stored audit records**] and [assignment: ***immediately alert the administrators by displaying a message at the remote management console when an administrative session exists for each of the defined administrative roles***] if the audit trail is full.

Dependencies: FAU_STG.1 Protected audit trail storage

Application Note: The TOE will overwrite "old" audit records once the audit trail is full. As with FAU_ARP.1, the TSF will also display a message at the remote console if an administrator that is already logged in, or when an administrator logs in. This requirement specifies that the message is sent to the first established session for each of the defined roles to ensure someone in the administrator staff is aware of the alert as soon as possible.

6.1.2 Class FCS: Cryptographic Support

The cryptography used in this product has not been FIPS certified nor has it been analyzed or tested to conform to cryptographic standards during this evaluation. All cryptography has only been asserted as tested by the vendor.

6.1.2.1 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 [assignment: ***RSA***] and specified cryptographic key sizes [assignment: ***2048 bits***] that meet the following: [assignment: ***RFC 2313***].

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

Application Note: This SFR supports key generation TLS v1.0.

6.1.2.2 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 [assignment: ***key zeroization***] that meets the following: [assignment: ***no standard***].

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

6.1.2.3 FCS_COP.1(1) Cryptographic operation

Hierarchical to: No other components.

FCS_COP.1.1 (1) The TSF shall perform [assignment: ***encryption and decryption***] in accordance with a specified cryptographic algorithm [assignment: ***AES***] and cryptographic key sizes [assignment: ***256 bits***] that meet the following: [assignment: ***RFC 3268***].

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or

FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4 Cryptographic key destruction

Application Note: This SFR supports encryption and decryption for TLS v1.0.

6.1.2.4 FCS_COP.1(2) Cryptographic operation

Hierarchical to: No other components.

FCS_COP.1.1(2) The TSF shall perform [assignment: ***cryptographic hashing services***] in accordance with a specified cryptographic algorithm [assignment: ***SHA-1*** *and* ***SHA-256***] and cryptographic key sizes [assignment: ***160 bits*** *and* ***256 bits***] that meet the following: [assignment: ***RFC 3174*** and ***FIPS 180-2***].

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

Application Note: This SFR supports cryptographic hashing services for TLS v1.0.

6.1.3 Class FDP: User Data Protection

6.1.3.1 FDP_IFC.1 Subset information flow control

Hierarchical to: No other components.

FDP_IFC.1.1 The TSF shall enforce the [assignment: ***UNAUTHENTICATED INFORMATION FLOW SFP***] on [assignment:

- ***source subject: TOE interface on which information is received;***
- ***destination subject: TOE interface to which information is destined;***
- ***information: network packets; and***
- ***operations: pass information].***

Dependencies: FDP_IFF.1 Simple security attributes

Application Note: In the PoliWall, the central issue is that there are two "subjects" (the sender of the packet (information) and the receiver of the packet) neither of which are under the control of the TOE. In order to use the FDP_IF requirements, we associate the potential set of subjects with a PoliWall interface. This makes sense because an administrator is able to determine what sets of IP addresses (for example) are associated with each of the physical PoliWall interfaces (assuming no other "backdoor"* *connectivity). Associating this potential set of subjects with an interface also allows the specification of subject attributes to be associated with something that is actually part of the TOE (the physical interface), as well as allow FDP_IFF.1.2 to be written so that it actually makes sense.*

Note that "operations" also is different from an operating-system-centric world because there is only one operation that the subjects really want: that the information is passed through the PoliWall.

6.1.3.2 FDP_IFF.1 Simple security attributes

Hierarchical to: No other components.

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

- ***a) Source subject security attributes: set of source subject identifiers***
- ***b) Destination subject security attributes: Set of destination subject identifiers***
- ***c) Information security attributes:***
    - ***presumed identity of source subject;***
    - ***identity of destination subject;***
    - ***transport layer protocol;***
    - ***services; destination subject service identifier (e.g., TCP or UDP destination port number);***
    - ***category code for external network traffic;***
    - ***Stateful packet attributes:***
        - ***i. Connection-oriented protocols:***
            1. ***sequence number,***
            2. ***acknowledgement number,***
            3. ***Flags:***
                - ***a. SYN;***
                - ***b. ACK;***
                - ***c. RST;***
                - ***d. FIN;***
                - ***e. PSH;***
                - ***f. URG;***
        - ***ii. Connectionless protocols:***
            1. ***source and destination network identifiers,***
            2. ***source and destination service identifiers.]***

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

- ***the presumed identity of the source subject is in the set of source subject identifiers;***

- ***the identity of the destination subject is in the set of source destination identifiers;***
- ***the information security attributes match the attributes in an information flow policy rule (contained in the information flow policy ruleset defined by the Security Administrator) according to the following algorithm [UNAUTHENTICATED INFORMATION FLOW SFP]; and***
- ***the selected information flow policy rule specifies that the information flow is to be permitted***].

FDP_IFF.1.3 The TSF shall enforce the [assignment: ***following stateful packet inspection rules:***

- ***whenever a packet is received that is not associated with an allowed established session (e.g., the SYN flag is set without the ACK flag being set), the information flow policy ruleset, as defined in FDP_IFF.1.2, is applied to the packet;***
- ***otherwise, the TSF associates a packet with an allowed established session***].

FDP_IFF.1.4 The TSF shall explicitly authorize an information flow based on the following rules: [assignment: ***when a flow has already been established and no changes to any policies have been made***].

FDP_IFF.1.5 The TSF shall explicitly deny an information flow based on the following rules: [assignment:

- ***the TOE shall reject requests for access or services where the presumed source ID of the information received by the TOE is not included in the set of source identifiers for the source subject***
- ***the TOE shall reject requests for access or services where the presumed source ID of the information received by the TOE is included in the list of source identifiers to be blocked by the REACT Server***].

Dependencies: FDP_IFC.1 Subset information flow control
FMT_MSA.3 Static attribute initialization

Application Note: Whenever a packet is received that is not associated with an allowed established session (e.g., the SYN flag is set without the ACK flag being set), the information flow policy ruleset, as defined in FDP_IFF.1.2(1), is applied to the packet; Otherwise, the TSF associates a packet with an allowed established session.

6.1.3.3 FDP_RIP.1(1) Subset residual information protection

Hierarchical to: No other components.

FDP_RIP.1.1 (1) The TSF shall ensure that any previous information content of a resource is made unavailable upon the [selection: **allocation of the resource to,**

deallocation of the resource from] the following objects: [assignment: ***kernel level objects***].

Dependencies: No dependencies.

6.1.3.4 FDP_RIP.1(2) Subset residual information protection

Hierarchical to: No other components.

FDP_RIP.1.1 (2) The TSF shall ensure that any previous information content of a resource is made unavailable upon the [selection: **deallocation of the resource from**] the following objects: [assignment: ***user-space program level***].

Dependencies: No dependencies.

6.1.4 Class FIA: Identification & Authentication

6.1.4.1 FIA_AFL.1 Authentication failure handling

Hierarchical to: No other components.

FIA_AFL.1.1 The TSF shall detect when [selection: **an administrator configurable positive integer within** [assignment: ***2-25***]] unsuccessful authentication attempts occur related to [assignment:

- ***administrators attempting to authenticate remotely***
- ***has a maximum authentication attempts of # before a user is locked out, where # is definable by the Security Administrator***
- ***has a lockout duration of # minutes, where # is definable by the Security Administrator***

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been [selection: **met**], the TSF shall [assignment:

- ***at the option of the Security Administrator prevent the remote administrators from performing activities that require authentication until an action is taken by the Security Administrator, or until a Security Administrator defined time period has elapsed***
- ***has a lockout duration of # minutes, where # is definable by the Security Administrator***
- ***has a maximum inactive session of # minutes before re-authentication is required, where # is definable by the Security Administrator***
- ***has a minimum session of # minutes before re-authentication is required, where # is definable by the Security Administrator***

Dependencies: FIA_UAU.1 Timing of authentication

6.1.4.2 FIA_ATD.1 User attribute definition

Hierarchical to: No other components.

FIA_ATD.1.1 The TSF shall maintain the following list of security attributes belonging to individual users: [assignment: ***username, password, certificate, role, security descriptor, Admin Session Policy***].

Dependencies: No dependencies.

6.1.4.3 FIA_SOS.2 TSF Generation of secrets

Hierarchical to: No other components.

FIA_SOS.2.1 The TSF shall provide a mechanism to generate secrets that meet [assignment:

- ***a # character minimum***
- ***at least # of the following 4 metrics: uppercase characters, lowercase characters, numbers, symbol, where # is definable by the Security Administrator***
- ***is not one of the previous # used passwords, where # is definable by the Security Administrator***
- ***has a maximum life of # days, where # is definable by the Security Administrator***
- ***has a minimum life of # days, where # is definable by the Security Administrator***
- ***has a maximum inactive session of # minutes before re-authentication is required, where # is definable by the Security Administrator***
- ***has a minimum session of # minutes before re-authentication is required, where # is definable by the Security Administrator***].

FIA_SOS.2.2 The TSF shall be able to enforce the use of TSF generated secrets for [assignment: ***authentication and access control***].

Dependencies: No dependencies.

6.1.4.4 FIA_UAU.1 Timing of authentication

Hierarchical to: No other components.

FIA_UAU.1.1 The TSF shall allow [assignment: ***ICMP if configured by the Security Administrator,***] on behalf of the user to be performed before the user is authenticated.

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

Dependencies: FIA_UID.1 Timing of identification

6.1.4.5 FIA_UAU.5 Multiple authentication mechanisms

Hierarchical to: No other components.

FIA_UAU.5.1 The TSF shall provide [assignment: ***username/password. and username/password with client certificate***] to support user authentication.

FIA_UAU.5.2 The TSF shall authenticate any user's claimed identity according to the [assignment: ***Security Administrators configurable settings***].

Dependencies: No dependencies.

6.1.4.6 FIA_UID.2 User identification before any action

Hierarchical to: FIA_UID.1 Timing of identification

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

Dependencies: No dependencies.

6.1.4.7 FIA_USB.1 User-subject binding

Hierarchical to: No other components.

FIA_USB.1.1 The TSF shall associate the following user security attributes with subjects acting on the behalf of that user: [assignment: ***all user attributes as specified in FIA_ATD.1***].

FIA_USB.1.2 The TSF shall enforce the following rules on the initial association of user security attributes with subjects acting on the behalf of users: [assignment: ***association of a user's attributes and role in a session object***].

FIA_USB.1.3 The TSF shall enforce the following rules governing changes to the user security attributes associated with subjects acting on the behalf of users: [assignment: ***Revocation of the user-subject binding and termination of the user's session under the following conditions:***

- ***Disabling of the user***
- ***Changes to the Admin Session Policy***
- ***Revocation of the role***].

Dependencies: FIA_ATD.1 User attribute definition

6.1.5 **Class FMT: Security Management**

6.1.5.1 FMT_MOF.1 Management of security functions behavior

Hierarchical to: No other components.

FMT_MOF.1.1 The TSF shall restrict the ability to [selection: **see Operation column of Table 6-3 Management Functions of the TOE**] the functions [assignment: ***see Object column of Table*** **6-3 Management Functions of the TOE**] to [assignment: ***See Role column of Table*** **6-3 Management Functions of the TOE**].

Dependencies: FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions

Application Note: The Object column contains a description of the TSF data that represents the information which can be changed by an administrator or external IT entity. For example, 'Audit Trail Threshold' represents the TSF data which contains the value that indicates when the audit trail has meet a percentage of the audit trail capacity, and therefore generates an alarm.

Object	Operation	Role (attribute)
Security Alarms - Alarm (FAU_ARP.1)	Disable, Enable	Security Administrator
Security Alarms – Auditable Alarm (FAU_ARP.1)	Disable, Enable	Security Administrator
Security Alarm Acknowledgement (FAU_ARP_EXT.1)	Modify the behavior of	Security Administrator, Audit Administrator, Cryptographic Administrator
Audit Trail Threshold (FAU_SAA.1, FAU_STG.3)	Determine the behavior of, Disable, Enable, Modify the	Security Administrator

Object	Operation	Role (attribute)
	behavior of	
Audit Trail (FAU_SAR.1, FAU_SAR.3)	Determine the behavior of, Disable, Enable, Modify the behavior of	Security Administrator, Audit Administrator, Cryptographic Administrator Read-Only
Audit Trail (FAU_STG.1)	Modify the behavior of	Audit Administrator
x509 Certificates, encryption setting (FCS_COP.1(1), FCS_COP.1(2), FCS_CKM.1, FCS_CKM.4)	Determine the behavior of, Disable, Enable, Modify the behavior of	Cryptographic Administrator
Information Flow Policy Rule (FDP_IFC.1, FDP_IFF.1)	Modify the behavior of	Security Administrator
Method of unlocking of locked accounts (FIA_AFL.1)	Modify the behavior of	Security Administrator
Password Policy (FIA_SOS.2, FTA_SSL.3)	Modify the behavior of	Security Administrator
ICMP (FIA_UAU.1)	Modify the behavior of , Disable, Enable	Security Administrator
Authentication method (FIA_UAU.5)	Modify the behavior of	Security Administrator
Time Stamp (FPT_STM.1)	Modify the behavior of	Authorized IT Entity (NTP Server)
TSF Self-Tests – Periodic Interval (FPT_TST.1)	Modify the behavior of	Security Administrator
TSF Self-Tests – Perform (FPT_TST.1)	Modify the behavior of	Security Administrator, Audit Administrator, Cryptographic Administrator
Quotas (FRU_RSA.1)	Modify the behavior of	Security Administrator
Banner (FTA_TAB.1)	Modify the behavior of	Security Administrator
Admin Session Policy (FTA_TSE.1)	Modify the behavior of	Security Administrator

Object	Operation	Role (attribute)
Users	Modify the behavior of	Security Administrator
Categories	Modify the behavior of	Security Administrator
Category Database	Modify the behavior of	Authorized IT Entity (Auto Update Server)

Table 6-3: Management Functions of the TOE

6.1.5.2 FMT_MSA.1 Management of security attributes

Hierarchical to: No other components.

FMT_MSA.1.1 The TSF shall enforce the [assignment: ***UNAUTHENTICATED INFORMATION FLOW SFP***] to restrict the ability to [selection: **change_default, query, modify**] the security attributes [assignment: ***referenced in the indicated policies***] to [assignment: ***the Security Administrator***].

Dependencies: [FDP_ACC.1 Subset access control, or
FDP_IFC.1 Subset information flow control]
FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions

Application Note: The attributes associated with stateful packet inspection are not expected to be managed by the Security Administrator.

6.1.5.3 FMT_MSA.3 Static attribute initialization

Hierarchical to: No other components.

FMT_MSA.3.1 The TSF shall enforce the [assignment: ***UNAUTHENTICATED INFORMATION FLOW SFP***] to provide [selection **restrictive**] default values for security attributes that are used to enforce the SFP.

Application Note: The security attributes to which this requirement refers, are the security attributes which define the default information flow policy ruleset, which is deny all network traffic.

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

Dependencies: FMT_MSA.1 Management of security attributes
FMT_SMR.1 Security roles

6.1.5.4 FMT_MTD.1 Management of TSF data

Hierarchical to: No other components.

FMT_MTD.1.1 The TSF shall restrict the ability to [***see Operation column of Table 6-4 Management of TSF Data***] the [assignment: ***see Object column of Table 6-4 Management of TSF Data***] to [assignment: ***see Role column of Table 6-4 Management of TSF Data**]*.

Dependencies: FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions

Application Note: *The Object column contains a description of the TSF data that represents the information which can be changed by an administrator or external IT entity. For example, 'Audit Trail Threshold' represents the TSF data which contains the value that indicates when the audit trail has meet a percentage of the audit trail capacity, and therefore generates an alarm.*

Object	Operation	Role (attribute)
Security Alarms - Alarm (FAU_ARP.1)	**selection: change_default**	Security Administrator
Security Alarms – Auditable Alarm (FAU_ARP.1)	**selection: change_default**	Security Administrator
Security Alarm Acknowledgement (FAU_ARP_EXT.1)	***assignment: Accept***	Security Administrator, Audit Administrator, Cryptographic Administrator
Audit Trail Threshold (FAU_SAA.1, FAU_STG.3)	**selection: change_default, query, modify**	Security Administrator
Audit Trail (FAU_SAR.1, FAU_SAR.3)	**selection: query**	Security Administrator, Audit Administrator, Cryptographic Administrator Read-Only
Audit Trail (FAU_STG.1)	**selection: delete**	Audit Administrator

Object	Operation	Role (attribute)
x509 Certificates, encryption (FCS_COP.1(1), FCS_COP.1(2), FCS_CKM.1, FCS_CKM.4)	***assignment: install, update, disable, enable, configure***	Cryptographic Administrator
Information Flow Policy Rule (FDP_IFC.1, FDP_IFF.1)	**selection: change_default, query, modify**	Security Administrator
Method of unlocking of locked accounts (FIA_AFL.1)	**selection: change_default**	Security Administrator
Password Policy (FIA_SOS.2, FTA_SSL.3)	**selection: change_default, query, modify**	Security Administrator
ICMP (FIA_UAU.1)	**selection: change_default, query, modify**	Security Administrator
Authentication method (FIA_UAU.5)	**selection: change_default**	Security Administrator
Time Stamp (FPT_STM.1)	**selection: modify**	Authorized IT Entity (NTP Server)
TSF Self-Tests – Periodic Interval (FPT_TST.1)	**selection: modify**	Security Administrator
TSF Self-Tests – Perform (FPT_TST.1)	***assignment: Run***	Security Administrator, Audit Administrator, Cryptographic Administrator
Quotas (FRU_RSA.1)	**selection: change_default, query, modify**	Security Administrator
Banner (FTA_TAB.1)	**selection: modify**	Security Administrator
Admin Session Policy (FTA_TSE.1)	**selection: change_default, query, modify, delete**	Security Administrator
Users	***assignment: Create*** **selection: query, modify, delete**	Security Administrator
Categories	***assignment: Create*** **selection: query, modify, delete**	Security Administrator
Category Database	***assignment: Update***	Authorized IT Entity
Configuration Information	***assignment: View***	Read-Only

Object	Operation	Role (attribute)
		Security Administrator, Audit Administrator, Cryptographic Administrator

Table 6-4: Management of TSF Data

6.1.5.5 FMT_MTD.2 Management of limits on TSF data

Hierarchical to: No other components.

FMT_MTD.2.1 The TSF shall restrict the specification of the limits for [assignment: ***quotas on transport-layer connections and controlled connection-oriented resources***] to [assignment: ***the Security Administrator***].

FMT_MTD.2.2 The TSF shall take the following actions, if the TSF data are at, or exceed the indicated limits: [assignment: ***drops all packets above the quota***].

Dependencies: FMT_MTD.1 Management of TSF data
FMT_SMR.1 Security roles

Application Note: The TOE assigns quotas based upon IP address and category code. Therefore, it makes quota decisions based upon the bandwidth of both the Transport Layer connections and controlled connection-oriented resources.

6.1.5.6 FMT_REV.1 Revocation

Hierarchical to: No other components.

FMT_REV.1.1 The TSF shall restrict the ability to revoke [assignment: ***security attributes***] associated with the [selection: **users**, [assignment: ***information flow policy ruleset, services available to unauthenticated users***]] under the control of the TSF to [assignment: ***the Security Administrator].***

FMT_REV.1.2 The TSF shall enforce the rules [assignment:

- ***revocation of a user's role (Security Administrator, Cryptographic Administrator, Audit Administrator);***
- ***changes to the Admin Session Policy;***
- ***disabling of the user;***
- ***changes to the information flow policy ruleset when applied;***
- ***disabling of a service available to unauthenticated users***]

Dependencies: FMT_SMR.1 Security roles

Application Note: The security attributes associated with users are defined in FIA_ATD.1; the intent is to include an indication that a user is allowed to act in a role (Security Administrator Cryptographic Administrator or Audit Administrator).
The security attributes associated with the information flow policy ruleset are the rules themselves, and any attributes listed in the FDP_IFF.1.1 elements that are grouped to create new attributes that can be used in forming a rule. The security attributes associated with the services available to unauthenticated users is just the list of services.

6.1.5.7 FMT_SMF.1 Specification of Management Functions

Hierarchical to: No other components.

FMT_SMF.1.1 The TSF shall be capable of performing the following management functions: [assignment: ***operations on objects as defined in the Object and Operation columns of Table 6-3 Management Functions of the TOE***].

Dependencies: No dependencies.

6.1.5.8 FMT_SMR.2 Restrictions on security roles

Hierarchical to: FMT_SMR.1 Security roles

FMT_SMR.2.1 The TSF shall maintain the roles: [assignment:

- ***Security Administrator,***
- ***Cryptographic Administrator (i.e. users authorized to perform cryptographic initialization and management functions),***
- ***Audit Administrator, and***
- ***Read-Only***].

FMT_SMR.2.2 The TSF shall be able to associate users with roles.

FMT_SMR.2.3 The TSF shall ensure that the conditions [assignment:

- ***all roles shall be able to administer the TOE remotely;***
- ***all default roles are distinct; that is, there shall be no overlap of operations performed by each default role, with the following exceptions:***
    - ***i. all administrators with a default role can review the audit trail;***
    - ii. ***all administrators with a default role can invoke the self-tests and***
    - iii. ***all administrators with a default role can accept alarms/acknowledgements***]

are satisfied.

Dependencies: FIA_UID.1 Timing of identification

6.1.6 **Class FPT: Protection of the TSF**

6.1.6.1 FPT_FLS.1 Failure with preservation of secure state

Hierarchical to: No other components.

FPT_FLS.1.1 The TSF shall preserve a secure state when the following types of failures occur: [assignment: ***when any number of the following modules goes down: Auto Update, PoliWall Process, msglogd, syslogd, pktlogd, pktlog6d***].

Dependencies: No dependencies

6.1.6.2 FPT_STM.1 Reliable time stamps

Hierarchical to: No other components.

FPT_STM.1.1 The TSF shall be able to provide reliable time stamps.

Dependencies: No dependencies.

6.1.6.3 FPT_TST.1 TSF testing

Hierarchical to: No other components.

FPT_TST.1.1 The TSF shall run a suite of self tests [selection: **during initial start-up, periodically during normal operation, at the request of the authorized user**] to demonstrate the correct operation of [selection: **the TSF**].

FPT_TST.1.2 The TSF shall provide authorized users with the capability to verify the integrity of [selection: **TSF data**].

FPT_TST.1.3 The TSF shall provide authorized users with the capability to verify the integrity of [selection: **TSF**].

Dependencies: No dependencies.

6.1.7 **Class FRU: Resource Utilization**

6.1.7.1 FRU_FLT.1 (1) Degraded fault tolerance

Hierarchical to: No other components.

FRU_FLT.1.1 (1) The TSF shall ensure the operation of [assignment: ***information flow control***] when the following failures occur: [assignment: ***Auto Update module goes down***].

Dependencies: FPT_FLS.1 Failure with preservation of secure state

6.1.7.2 FRU_FLT.1 (2) Degraded fault tolerance

Hierarchical to: No other components.

FRU_FLT.1.1 (2) The TSF shall ensure the operation of [assignment: ***remote administration functions and access control***] when the following failures occur: [assignment: ***PoliWall Process module goes down***].

Dependencies: FPT_FLS.1 Failure with preservation of secure state

6.1.7.3 FRU_FLT.1 (3) Degraded fault tolerance

Hierarchical to: No other components.

FRU_FLT.1.1(3) The TSF shall ensure the operation of [assignment: ***auditing functions***] when the following failures occur: [assignment: ***when any number of the following auditing modules go down: msglogd, syslogd, pktlogd, pktlog6d***].

Dependencies: FPT_FLS.1 Failure with preservation of secure state

6.1.7.4 FRU_FLT.2 Limited fault tolerance

Hierarchical to: FRU_FLT.1 Degraded fault tolerance

FRU_FLT.2.1 The TSF shall ensure the operation of all the TOE's capabilities when the following failures occur: [assignment: ***when any number of the following modules go down: Auto Update, PoliWall Process, msglogd, syslogd, pktlogd, pktlog6d***].

Dependencies: FPT_FLS.1 Failure with preservation of secure state

6.1.7.5 FRU_PRS.1 Limited priority of service

Hierarchical to: No other components.

FRU_PRS.1.1 The TSF shall assign a priority to each subject in the TSF.

FRU_PRS.1.2 The TSF shall ensure that each access to [assignment: ***UNAUTHENTICATED FLOW CONTROL***] shall be mediated on the basis of the subjects assigned priority.

Dependencies: No dependencies.

6.1.7.6 FRU_RSA.1 Maximum quotas

Hierarchical to: No other components.

FRU_RSA.1.1 The TSF shall enforce maximum quotas of the following resources: [assignment: ***transport layer representation, controlled connection-oriented resources***] that [selection: **subjects**] can use [selection: **simultaneously**].

Dependencies: No dependencies.

Application Note: This requirement has been included to capture the TOE's ability to allow Security Administrator's to assign quotas based on bandwidth to network traffic associated with a category code. Once the network traffic for a particular category code exceeds the quota all packets which exceed that quota will be dropped.

6.1.8 **Class FTA: TOE Access**

6.1.8.1 FTA_SSL.3 TSF-initiated termination

Hierarchical to: No other components.

FTA_SSL.3.1 The TSF shall terminate an interactive session after a [assignment: ***Security Administrator-configurable time interval of session inactivity***].

Dependencies: No dependencies.

Application Note: The term "session" used in this requirement refers to an administrator's remote session.

6.1.8.2 FTA_TAB.1 Default TOE access banners

Hierarchical to: No other components.

FTA_TAB.1.1 Before establishing a user session, the TSF shall display an advisory warning message regarding unauthorized use of the TOE.

Dependencies: No dependencies.

Application Note: The access banner applies whenever the TOE will provide a prompt for identification and authentication (e.g., administrators). The intent of this requirement is to advise users of warnings regarding the unauthorized use of the TOE and to provide the Security Administrator with control over what is displayed (e.g., if the Security Administrator chooses, they can remove banner information that informs the user of the product and version number).

6.1.8.3 FTA_TSE.1 TOE session establishment

Hierarchical to: No other components.

FTA_TSE.1.1 The TSF shall be able to deny session establishment based on [assignment: ***Admin Session Policy***].

Dependencies: No dependencies.

Application Note: The term "session" used in this requirement refers to an administrator's remote session.

Application Note: Admin session policy is based on the source restriction (e.g. IP addresses), time, and day.

6.1.9 **Class FTP: Trusted Path**

6.1.9.1 FTP_ITC.1 Inter-TSF trusted channel

Hierarchical to: No other components.

FTP_ITC.1.1 The TSF shall provide a communication channel between itself and another trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.

FTP_ITC.1.2 The TSF shall permit [selection: **the TSF**] to initiate communication via the trusted channel.

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for [assignment: ***update of system time, SNMP, Category Code Database, PCELs and Remote Management Console].***

Dependencies: No dependencies.

6.1.9.2 FTP_TRP.1 Trusted path

Hierarchical to: No other components.

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

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

FTP_TRP.1.3 The TSF shall require the use of the trusted path for [selection: **initial user authentication**, [assignment: ***all administrative actions***]].

6.2 Operations Defined

The notation, formatting, and conventions used in this security target (ST) are consistent with version 3.1 of the Common Criteria for Information Technology Security Evaluation. All of the components in this ST are taken directly from Part 2 of the CC except the ones noted with "_EXT" in the component name. Font style and clarifying information conventions were developed to aid the reader.

The CC permits four functional component operations: assignment, iteration, selection, and refinement to be performed on functional requirements. These operations are defined in Common Criteria, Part 1 as:

6.2.1 **Assignments Made**

An assignment allows the specification of parameters and is specified by the ST author in [***italicized bold text***].

6.2.2 **Iterations Made**

An iteration allows a component to be used more than once with varying operations and are identified with the iteration number within parentheses after the short family name.

6.2.3 **Selections Made**

A selection allows the specification of one or more items from a list and is specified by the ST author in [**bold text**].

6.2.4 **Refinements Made**

A refinement allows the addition of details and is identified with "Refinement:" right after the short name. ~~The old text is shown with a strikethrough~~ and ***the new text is specified by*** ***italicized bold and underlined text***.

7 Security Assurance Requirements

This section identifies the Security Assurance Requirement components met by the TOE. These assurance components meet the requirements for EAL4 augmented with ALC_FLR.2 and ASE_TSS.2.

7.1 Security Architecture

7.1.1 Security Architecture Description (ADV_ARC.1)

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

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

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

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

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

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

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

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

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

7.1.2 Functional Specification with Complete Summary (ADV_FSP.4)

ADV_FSP.4.1D The developer shall provide a functional specification.

ADV_FSP.4.2D The developer shall provide a tracing from the functional specification to the SFRs.

ADV_FSP.4.1C The functional specification shall completely represent the TSF.

ADV_FSP.4.2C The functional specification shall describe the purpose and method of use for all TSFI.

ADV_FSP.4.3C The functional specification shall identify and describe all parameters associated with each TSFI.

ADV_FSP.4.4C The functional specification shall describe all actions associated with each TSFI.

ADV_FSP.4.5C The functional specification shall describe all direct error messages that may result from an invocation of each TSFI.

ADV_FSP.4.6C The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification.

ADV_FSP.4.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_FSP.4.2E The evaluator shall determine that the functional specification is an accurate and complete instantiation of the SFRs.

7.1.3 Implementation Representation of the TSF (ADV_IMP.1)

ADV_IMP.1.1D The developer shall make available the implementation representation for the entire TSF.

ADV_IMP.1.2D The developer shall provide a mapping between the TOE design description and the sample of the implementation representation. Content and presentation elements:

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

ADV_IMP.1.2C The implementation representation shall be in the form used by the development personnel.

ADV_IMP.1.3C The mapping between the TOE design description and the sample of the implementation representation shall demonstrate their correspondence. Evaluator action elements:

ADV_IMP.1.1E The evaluator *shall confirm* that, for the selected sample of the implementation representation, the information provided meets all requirements for content and presentation of evidence.

7.1.4 Architectural Design (ADV_TDS.3)

ADV_TDS.3.1D The developer shall provide the design of the TOE.

ADV_TDS.3.2D The developer shall provide a mapping from the TSFI of the functional specification to the lowest level of decomposition available in the TOE design.

ADV_TDS.3.1C The design shall describe the structure of the TOE in terms of subsystems.

ADV_TDS.3.2C The design shall describe the TSF in terms of modules.

ADV_TDS.3.3C The design shall identify all subsystems of the TSF.

ADV_TDS.3.4C The design shall provide a description of each subsystem of the TSF.

ADV_TDS.3.5C The design shall provide a description of the interactions among all subsystems of the TSF.

ADV_TDS.3.6C The design shall provide a mapping from the subsystems of the TSF to the modules of the TSF.

ADV_TDS.3.7C The design shall describe each SFR-enforcing module in terms of its purpose and relationship with other modules.

ADV_TDS.3.8C The design shall describe each SFR-enforcing module in terms of its SFR-related interfaces, return values from those interfaces, interaction with other modules and called SFR-related interfaces to other SFR-enforcing modules.

ADV_TDS.3.9C The design shall describe each SFR-supporting or SFR-non-interfering module in terms of its purpose and interaction with other modules.

ADV_TDS.3.10C The mapping shall demonstrate that all TSFIs trace to the behavior described in the TOE design that they invoke.

ADV_TDS.3.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_TDS.3.2E The evaluator shall determine that the design is an accurate and complete instantiation of all security functional requirements.

7.2 Guidance Documents

7.2.1 Operational User Guidance (AGD_OPE.1)

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

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

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

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

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

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

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

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

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

7.2.2 Preparative Procedures (AGD_PRE.1)

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

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

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

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

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

7.3 Lifecycle Support

7.3.1 Authorization Controls (ALC_CMC.4)

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

ALC_CMC.4.2D The developer shall provide the CM documentation.

ALC_CMC.4.3D The developer shall use a CM system.

ALC_CMC.4.1C The TOE shall be labeled with its unique reference.

ALC_CMC.4.2C The CM documentation shall describe the method used to uniquely identify the configuration items.

ALC_CMC.4.3C The CM system shall uniquely identify all configuration items.

ALC_CMC.4.4C The CM system shall provide automated measures such that only authorized changes are made to the configuration items.

ALC_CMC.4.5C The CM system shall support the production of the TOE by automated means.

ALC_CMC.4.6C The CM documentation shall include a CM plan.

ALC_CMC.4.7C The CM plan shall describe how the CM system is used for the development of the TOE.

ALC_CMC.4.8C The CM plan shall describe the procedures used to accept modified or newly created configuration items as part of the TOE.

ALC_CMC.4.9C The evidence shall demonstrate that all configuration items are being maintained under the CM system.

ALC_CMC.4.10C The evidence shall demonstrate that the CM system is being operated in accordance with the CM plan.

ALC_CMC.4.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.3.2 CM Scope (ALC_CMS.4)

ALC_CMS.4.1D The developer shall provide a configuration list for the TOE.

ALC_CMS.4.1C The configuration list shall include the following: the TOE itself; the evaluation evidence required by the SARs; the parts that comprise the TOE; the implementation representation; and security flaw reports and resolution status.

ALC_CMS.4.2C The configuration list shall uniquely identify the configuration items.

ALC_CMS.4.3C For each TSF relevant configuration item, the configuration list shall indicate the developer of the item.

ALC_CMS.4.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

7.3.3 Delivery Procedures (ALC_DEL.1)

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

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

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

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

7.3.4 Identification of Security Measures (ALC_DVS.1)

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

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

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

ALC_DVS.1.2E The evaluator shall confirm that the security measures are being applied.

7.3.5 Life-cycle Definition (ALC_LCD.1)

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

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

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

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

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

7.3.6 Tools and techniques (ALC_TAT.1)

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

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

ALC_TAT.1.1C Each development tool used for implementation shall be well-defined.

ALC_TAT.1.2C The documentation of each development tool shall unambiguously define the meaning of all statements as well as all conventions and directives used in the implementation.

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

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

7.3.7 Flaw reporting procedures (ALC_FLR.2)

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

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

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

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

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

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

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

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

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

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

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

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

7.4 Security Target Evaluation

7.4.1 Conformance Claims (ASE_CCL.1)

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

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

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

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

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

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

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

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

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

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

7.4.2 Extended Components Definition (ASE_ECD.1)

ASE_ECD.1.1D The developer shall provide a statement of security requirements.

ASE_ECD.1.2D The developer shall provide an extended components definition.

ASE_ECD.1.1C The statement of security requirements shall identify all extended security requirements.

ASE_ECD.1.2C The extended components definition shall define an extended component for each extended security requirement.

ASE_ECD.1.3C The extended components definition shall describe how each extended component is related to the existing CC components, families, and classes.

ASE_ECD.1.4C The extended components definition shall use the existing CC components, families, classes, and methodology as a model for presentation.

ASE_ECD.1.5C The extended components shall consist of measurable and objective elements such that conformance or nonconformance to these elements can be demonstrated.

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

ASE_ECD.1.2E The evaluator shall confirm that no extended component can be clearly expressed using existing components.

7.4.3 ST Introduction (ASE_INT.1)

ASE_INT.1.1D The developer shall provide an ST introduction.

ASE_INT.1.1C The ST introduction shall contain an ST reference, a TOE reference, a TOE overview and a TOE description.

ASE_INT.1.2C The ST reference shall uniquely identify the ST.

ASE_INT.1.3C The TOE reference shall identify the TOE.

ASE_INT.1.4C The TOE overview shall summarize the usage and major security features of the TOE.

ASE_INT.1.5C The TOE overview shall identify the TOE type.

ASE_INT.1.6C The TOE overview shall identify any non-TOE hardware/software/firmware required by the TOE.

ASE_INT.1.7C The TOE description shall describe the physical scope of the TOE.

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

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

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

7.4.4 Security Objectives (ASE_OBJ.2)

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

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

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

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

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

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

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

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

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

7.4.5 Security Requirements (ASE_REQ.2)

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

ASE_REQ.2.2D The developer shall provide a security requirements rationale.

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

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

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

ASE_REQ.2.4C All operations shall be performed correctly.

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

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

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

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

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

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

7.4.6 Security Problem Definition (ASE_SPD.1)

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

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

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

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

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

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

7.4.7 TOE Summary Specification (ASE_TSS.2)

ASE_TSS.2.1D The developer shall provide a TOE summary specification.

ASE_TSS.2.1C The TOE summary specification shall describe how the TOE meets each SFR.

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

ASE_TSS.2.2C The TOE summary specification shall describe how the TOE protects itself against interference and logical tampering.

ASE_TSS.2.2E The evaluator *shall confirm* that the TOE summary specification is consistent with the TOE overview and the TOE description.

ASE_TSS.2.3C The TOE summary specification shall describe how the TOE protects itself against bypass.

7.5 Tests

7.5.1 Analysis of Coverage (ATE_COV.2)

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

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

ATE_COV.2.2C The analysis of the test coverage shall demonstrate that all TSFIs in the functional specification have been tested.

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

7.5.2 Basic Design (ATE_DPT.2)

ATE_DPT.2.1D The developer shall provide the analysis of the depth of testing.

ATE_DPT.2.1C The analysis of the depth of testing shall demonstrate the correspondence between the tests in the test documentation and the TSF subsystems and SFR-enforcing modules in the TOE design.

ATE_DPT.2.2C The analysis of the depth of testing shall demonstrate that all TSF subsystems in the TOE design have been tested.

ATE_DPT.2.3C The analysis of the depth of testing shall demonstrate that the SFR-enforcing modules in the TOE design have been tested.

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

7.5.3 Functional Tests (ATE_FUN.1)

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

ATE_FUN.1.2D The developer shall provide test documentation

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

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

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

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

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

7.5.4 Independent Testing (ATE_IND.2)

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

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

ATE_IND.2.2C The developer shall provide an equivalent set of resources to those that were used in the developer's functional testing of the TSF.

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

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

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

7.6 Vulnerability Assessment

7.6.1 Vulnerability Analysis (AVA_VAN.3)

AVA_VAN.3.1D The developer shall provide the TOE for testing.

AVA_VAN.3.1C The TOE shall be suitable for testing.

AVA_VAN.3.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_VAN.3.2E The evaluator shall perform a search of public domain sources to identify potential vulnerabilities in the TOE.

AVA_VAN.3.3E The evaluator shall perform an independent, focused vulnerability analysis of the TOE using the guidance documentation, functional specification, TOE design, security architecture description and implementation representation to identify potential vulnerabilities in the TOE.

AVA_VAN.3.4E The evaluator shall conduct penetration testing, based on the identified potential vulnerabilities, to determine that the TOE is resistant to attacks performed by an attacker possessing Enhanced-Basic attack potential.

8 TOE Summary Specification

The following sections identify the security functions of the TOE. They include Security Audit, Cryptographic Support, Identification and Authentication, User Data Protection, Security Management, Protection of the TSF, Trusted Path/Channel, Resource Utilization, and TOE Access.

8.1 Security Audit

8.1.1 Audit Logs

Audit records will be generated only for specified (auditable) events: start-up and shutdown of audit functions, all auditable events for the specified level of audit, as listed in Table 6-2 Auditable Events. The TOE records the (1) date and time of the event, (2) type of event, (3) subject identity (if applicable), and the outcome of the event (success or failure) within each audit record. Each auditable event will be associated with the identity of the user who caused the event to occur. If an Administrator fails to authenticate with the correct credentials, no user association will be required because they are not yet under the TOE's control (one must be an authenticated Administrator in order for this to take place).

All audit data should be capable of being searched and sorted across all specified criteria. Only the user identity attribute is used to search and sort the audit logs for Administrator actions. The command type is only used to search the audit logs for Administrator actions. The date and time attributes are only used to search and sort the audit logs for flow control decisions.

The TOE provides the following audit logs:

- **IPv4 Packet Logs** - Data for all dropped IPv4 packets by source IP, destination IP, protocol, cause and country.
- **IPv6 Packet Logs** - Data for all dropped IPv6 packets by source IP, destination IP, protocol, cause and country.
- **Message Logs** - Shows system information, warning and error messages.
- **Command Logs** - System commands executed by TOE administrators.

The following table displays an IPv4/IPv6 Packet Log:

Feature	Description
Find By	The dropdown list provides options to find information by date, text string, or line number. Select an option, enter the query in the adjacent text box and click "Go". This moves the log to the first record that matches the query.
Lines Per Page	Enter the number of lines of data to display on each page of the log report. The default is 25.

Line Number	Line numbers are assigned by the system. Click on a line number to position the entry at the top of the page.
Date/Time	Timestamp of the associated event
First | Previous | Next | Last	Page navigation buttons
Protocol	Identifies the protocol in the IP header of the packet
Flags	Indicates the TCP flags that are set in the rejected packet, which may include any combination of FIN, SYN, RST, PSH, ACK, and URG. This field will be blank if the protocol is something other than TCP.
Cause	Indicates the filtering mechanism responsible for dropping the packet; Country Map, Override or Exception. Click on the Magnifying Glass icon to invoke the Packet Evaluator tool.
Country	Country associated with the packet's IP source IP address.

Table 8-1: Packet Log Contents

Feature	Description
Find By	The dropdown list provides options to find information by date, text string or line number. Select an option, enter the query in the adjacent text box and click "Go". This moves the log to the first record that matches the query.
Filter By	Data can be filtered by text, user or role. The input area will change according to the selected option to reveal available roles, users or a query input area as required. This alters the log to only display matching records.
Lines Per Page	Enter the number of lines of data to display on each page of the log report. The default is 25.
Line Number	Line numbers are assigned by the system. Click on a line number to position the entry at the top of the page.
Date/Time	Timestamp of the associated event
Millisecond	Fraction of a second timestamp of the associated event.
Message	Events details
First | Previous | Next | Last	Page navigation buttons
Protocol	Identifies the protocol in the IP header of the packet

Table 8-2: System Log Contents

The following table displays a Command Log:

Feature	Description
Filter By	Data can be filtered by text, user, or command type.

	This input area will change according to the selected option to reveal available users, command types, or a query input as required. This alters the log to only display matching records
Lines Per Page	Enter the number of lines of data to display on each page of the log report. The default is 25.
Line Number	Line numbers are assigned by the system. Click on a line number to position the entry at the top of the page.
Date/Time	Timestamp of the associated event
Millisecond	Fraction of a second timestamp of the associated event
Message	Event details
First | Previous | Next | Last	Page navigation buttons
User Name	User who issued the command
Role	Role of Administrator who issued command
Type	Type of command
Command	The command issued

Table 8-3: Command Log Contents

It is the TOE's responsibility to preserve the stored audit records from the audit trail from unauthorized deletion and protecting its integrity by preventing modifications to it. The only Administrator who is authorized to delete audit records is the Audit Administrator; however no Administrator is authorized to modify audit records. The Audit Administrator will delete audit records by specifying a percentage of the IPv4 Packet Log, IPv6 Packet Log, or Message Log to purge, and that percentage of the log file will be purged, starting from the beginning of the specified log.

If the audit trail is full, the TOE has the ability to overwrite older audit records. It will do this by overwriting the oldest audit records first, 1 page (4096 bytes) of records at a time. Once critical mass has been reached though, a message will be sent to any currently connected administrator's remote console notifying them of such an event.

8.1.2 Security Alarms & Violations

The TOE has the ability to display alarm messages to the administrator, identifying the potential security violation and making the audit record contents accessible that are associated with the auditable event(s) that generated the alarm. These alarm messages, which are produced by the PoliWall Process, are displayed at the remote console if an administrator is already logged in, or when an administrator logs in if the alarm message has not been acknowledged. Additionally, the TOE provides an auditable alarm that can be configured to sound an alarm if desired by the Security Administrator.

Acknowledgement messages will be immediately displayed by the TSF at all remote administrator sessions that received the alarm and identifying the reference to the potential security violation, a notice that the alarm has indeed been acknowledged, the timestamp of its acknowledgement, and the user identifier that acknowledged the alarm.

Rules will be put in place to monitor audited events, and with these rules potential violations of the enforcement of the security requirements will be indicated. Once a threshold has been set for a particular event, an alarm will be generated. Upon begin generated, the “count” for that event will be reset to zero. The Security Administrator is the sole individual responsible for specifying the limit for authentication failures as well as the threshold for the audit trail - which, if occur, will generate an alarm indicating a potential security violation.

The following rules apply to data pertaining to or extracted from the audit trail:

- All Administrators have the ability to read data from the audit trail, with the exception of those prohibited from reading such data. That data must be presented in an interpretable fashion for the Administrator(s) viewing it.
- Searching and sorting of the audit data is permitted based on user identity, command type and a range of one or more or both of dates and times.
- Audit log data should be protected against unauthorized deletion (the Audit Administrator is the only Administrator allowed to delete records) and/or modifications to the records contained in the audit trail (no Administrator is authorized to make modifications to audit records).
- Whenever an administrative session exists for the Security Administrator, Audit Administrator, and Cryptographic Administrator, the administrators will be immediately alerted at the remote management console with the receipt of an alarm. The administrator must be logged in or be in the process of logging in order to receive this message. The alarm message will sent be sent to all established sessions for each of the Administrators so that this notification is made known as soon as possible.
- If the audit trail’s threshold has been reached and is full, the oldest stored audit records will be overwritten. Once this occurs a message will be sent to the remote management console notifying of such an occurrence.
- The Audit Administrators will have the option of purging data from the logs by specifying a percentage of the records to be deleted. This will delete the oldest records from the specified log.

8.2 Cryptographic Support

The TOE provides for cryptography to be used between itself and other entities to which it is connected. Encryption is used between the TOE and the web interface, for Auto-Update, IPsec, NTP, SNMP, and for communications with the Remote Management Console. Specifically, the TOE allows for the generation, destruction, and encryption of keys. The cryptographic keys are overwritten with a single overwrite of pseudo-randomly generated bits to zeroize out the memory.

- Encryption between the TOE and the web interface (SSL for the https connection)
- Encryption for Auto-Update
- Encryption for IPSEC
- Encryption for NTP
- Encryption for SNMP
- Encryption for REACT
- Encryption for Remote Management Console (RMC) Server
- Encryption for communications with Admin Web GUI:

Encryption for the TOE and the aforementioned interfaces will be as follows:

Purpose	**Usage**	**Algorithm**	**Size**	**Standard**
Key Generation	[redacted]	RSA	2048	RFC 2313
Key Destruction	[redacted]	Key Zeroization	[redacted]	No Standard.
Crypto Operation (1)	Encryption/decryption	AES	256	RFC 3268
Crypto Operation (2)	Cryptographic Hashing	SHA-1	160	RFC 3174
Crypto Operation (3)	Cryptographic Hashing	SHA-256	256	FIPS 180-2

OpenSSL-FIPS version 1.2 is used by the TOE.

The cryptography library used in this product has been FIPS certified. The implementation of the cryptographic library is vendor-asserted, not FIPS-asserted.

8.3 Identification and Authentication

In order to authenticate to the TOE and perform TOE processes, users must either enter (1) their username and password or (2) username, password, and client certificate. The Security Administrator will define which of the authentication methods are to be used. These attributes, along with role, security descriptor and admin session policy, are maintained by a database within the TOE. When an administrator unsuccessfully attempts to authenticate to the TOE a given amount of times that account will be locked until further notice. The threshold for the number of times will be set by the Security Administrator and must be between 2 and 25 attempts. There are two ways that an account can be unlocked – either by the Security Administrator or when the specified time from the account locking has elapsed. If authentication and identification has been successfully completed, the Administrator's functions associated with the role will be displayed/granted.

Password Policy

The TOE comes preconfigured with mechanisms for creating a password and strictly enforces them. The mechanisms put in place for password creation are:

- must be a # character minimum, where # is definable by the Security Administrator
- must be at least # of the following 4 metrics: uppercase characters, lowercase characters, numbers, symbol, where # is definable by the Security Administrator
- is not one of the previous # used passwords, where # is definable by the Security Administrator
- has a maximum life of # days, where # is definable by the Security Administrator
- has a minimum life of # days, where # is definable by the Security Administrator
- has a maximum authentication attempts of # before an Administrator is locked out, where # is definable by the Security Administrator
- has a lockout duration of # minutes, where # is definable by the Security Administrator
- has a maximum inactive session of # minutes before re-authentication is required, where # is definable by the Security Administrator
- has a minimum session of # minutes before re-authentication is required, where # is definable by the Security Administrator

The only action which is permitted to be performed without authenticating to the TOE is ICMP (ping). This is wholly up to the discretion of the Security Administrator whether or not they will allow this action to be enabled or disabled without authenticating to the TOE; all other TOE actions require Administrators to properly authenticate to the TOE.

The TOE allows for the association of an Administrator's security attributes to be attributed to the Administrator acting on their behalf; the rules governing this association of attributes and the changing of those attributes will be strictly enforced by the Security Administrator.

The TOE contains the following processes that require authentication:

- PoliWall Process
- Network Time Protocol (NTP) Server
  - NTP assures accurate synchronization of computer clock times in a network of computers. NTP synchronizes the PoliWall's clock with the specified servers. PoliWall's NTP servers are set by choosing up to three IPv4 and/or IPv6 servers.
- SNMP
  - Will be used to monitor network-attached devices for conditions that warrant administrative attention
  - Administrators will be able to poll the TOE to gather statistics for the traffic flowing through the TOE
  - SNMP traps can be sent out to a specified external server when certain events occur (e.g. alert being raised)
- RMC Server
  - Will be used to query for commands to run on the TOE as if a Remote Admin was running the command.
  - The TOE will use SSL to authenticate the remote endpoint
- REACT Server
  - Will be used to specify IP addresses to be blocked from passing traffic through the TOE.
  - The TOE will use SSL and RSA signatures to authenticate the remote endpoint

The TOE has policies that govern its logon process for the aforementioned areas, which are:

- The number of failed login attempts is 5 by default and can be set as high as 25 if desired. Administrator accounts will be locked after the maximum number of failed login attempts is exceeded. Security Administrators can unlock the accounts or Administrators can try again after the “Minutes Until Locked Account is Unlocked” time passes.
- Sessions will expire after 60 minutes of inactivity by default. The timeout can be set to a value between 2 and 60 minutes
- If an Administrator exceeds the maximum login attempts, the account will be locked for 30 minutes by default before additional login attempts can be made. The value can be set to a value between 5 and 1440 minutes
- The maximum session length is 480 minutes by default and can be set to a value between 15 and 1440 minutes, after which, active sessions will be disconnected.
- Passwords expire in 45 days by default. A value between 15 and 180 days can be set.

- Administrators are prevented from using one of their last 3 passwords when prompted to change their password. The setting can be increased to 10 if desired.

The Security Administrator is the sole entity that has the authority to allow non-authenticated Administrators access to the TOE. If the Security Administrator does grant access to the TOE, the only capability the unauthenticated Administrator will have is ping (ICMP). Otherwise, all Administrators must be properly authenticated before any other TSF actions are made. Once authenticated, an Administrator will have all attributes associated with its role, e.g. Security Administrators have full access rights to the TOE with the exception of all privileges Audit and Cryptographic Administrators possess.

The Security Administrator may cause administration session for other users to be terminated in two different ways. First, the Security Administrator may terminate any other administration session on the TOE. Second, the Security Administrator may disable a user's account, which will cause all of that user's sessions to be terminated.

Once an Administrator has been correctly associated to a role, the information is stored in the session object (a system-level object in the PoliWall process) that is created for each authenticated session in the PoliWall process. This object contains the current Administrator and role for the session. The session identifier is passed in with the XML-messages for the function calls.
Additionally, PHP also stores tracks sessions by setting a cookie on the client. When the web browser presents the cookie, PHP can identify its session in that manner and have the session ID for the XML-message stored internally to be passed in.

When the TOE has entered Maintenance Mode, an Administrator may log in directly to the TOE and interact via a command-line based, menu-driven interface. The functionality of this interface is only accessible when the TOE is in Maintenance Mode. The logins for this mode are not user based, but rather are role-based. There are logins for Security Administrator, Audit Administrator, and Cryptographic Administrator. Each role-named login has an associated password that may be set through the Graphical User Interface, and the authentication is required to perform functions in Maintenance Mode. Maintenance mode will only be used when the TOE is not in its fully operational state. It is used to gain access to the TOE to address the fault(s) that placed it into Maintenance Mode. The credentials for this mode are set for the Administrators (1 per Administrator role) that provide access to a limited subset of the functionality defined for the specific Administrator and should not be shared outside of the users that have these Administrator privileges.

8.4 Management Functions (User Data Protection)

The TOE provides for enforcement of the Unauthenticated Information Flow SFP based on:

- Source Subject
    - TOE interface on which information is received
- Destination Subject
    - TOE interface to which information is destined
- Information
    - Network Packets
- Operations
    - Pass Information

The TOE, at a given time, will have two "subjects", the sender of the packets and the receiver of the packets "transparently bridged"; because of this transparent bridging, they are not under control of the TOE. An association is made with these subjects to the TOE because it can be readily determined what sets of IP addresses are associated with each of the TOE's interfaces. Using this association, the specification of security attributes with something that is a part of the TOE can be allowed.

The TOE provides for enforcement of the Unauthenticated Information Flow SFP based on the following subject and information security attributes:

- Source subject security attributes: set of source subject identifiers
- Destination subject security attributes: Set of destination subject identifiers
- Information security attributes:
    - presumed identity of source subject;
    - identity of destination subject;
    - transport layer protocol;
    - services; destination subject service identifier (e.g., TCP or UDP destination port number);
    - category code for external network traffic;
    - Stateful packet attributes:
        i. Connection-oriented protocols:
            1. sequence number,
            2. acknowledgement number,
            3. Flags:
                a. SYN;
                b. ACK;
                c. RST;
                d. FIN;
                e. PSH;
                f. URG;
        ii. Connectionless protocols:

1. source and destination network identifiers,

source and destination service identifiers.

Information is permitted to flow between a controlled source subject and controlled destination subject via a controlled operation so long as (1) the presumed identity of the source subject is in the set of source subject identifiers, (2) the identity of the destination subject is in the set of source destination identifiers, (3) the information security attributes match the attributes in the information flow policy rule, and (4) the selected information flow policy rule specifies that the information flow is to be permitted.

Packets should be inspected whenever a packet is received is not associated with an allowed established session and/or the information flow policy ruleset. In any other case, the packet should be associated with an allowed established session.

The information flow will be authorized when a flow has already been established and no changes to any policies have been made. In other words, once a session has been authorized and established, that authorization will persist until a new set of rules has been applied. The information flow will be rejected if the request for access or services where the presumed source ID of the information received by the TOE is not included in the set of source identifiers for the source subject. When a packet not related to an allowed established session is received, the information flow policy ruleset is applied to the subject. In all other cases, the packet is associated with an allowed established session. Any previous information content of a resource should be made available upon the allocation or reallocation of the resource from the list of objects.

8.4.1 **Access Control**

The TOE has several roles and has the following rules associated with them:

1. Security Administrator – has the ability to perform all functions except the ability to manage cryptography and delete audit logs
2. Audit Administrator – has the ability to delete audit records
3. Cryptographic Administrator – Manages all cryptographic functionality
4. Read-Only - Can only read configuration items and cannot change any configuration

User accounts will always be allowed to access the Read-Only role. The Security Administrator may assign and revoke the Security Administrator, Audit Administrator, and Cryptographic Administrator to any users. There must always be a user assigned to the Security Administrator role.

When users log in, they first start as the Read-Only role. From there, they may switch to any of the other roles they have access to. Re-entry of password is required to switch roles.

8.4.2 Flow Control

The TOE enforces the Unauthenticated Information Flow Control SFP to restrict the ability to change, default, and query or modify the security attributes to the Security Administrator. Manipulation of these security attributes can be used to create additional attributes that may be used in specifying information flow policy rules. This requirement is strictly limited to the Security Administrator.

The Unauthenticated Information Flow Control SFP must also provide restrictive values for security attributes to be used to enforce the SFP (i.e. deny all network traffic). The Security Administrator is the only Administrator with the ability to specify alternative initial values to override the aforementioned default values when an object/information is being created.

The following chart illustrates how information is allowed (or denied) through the TOE:

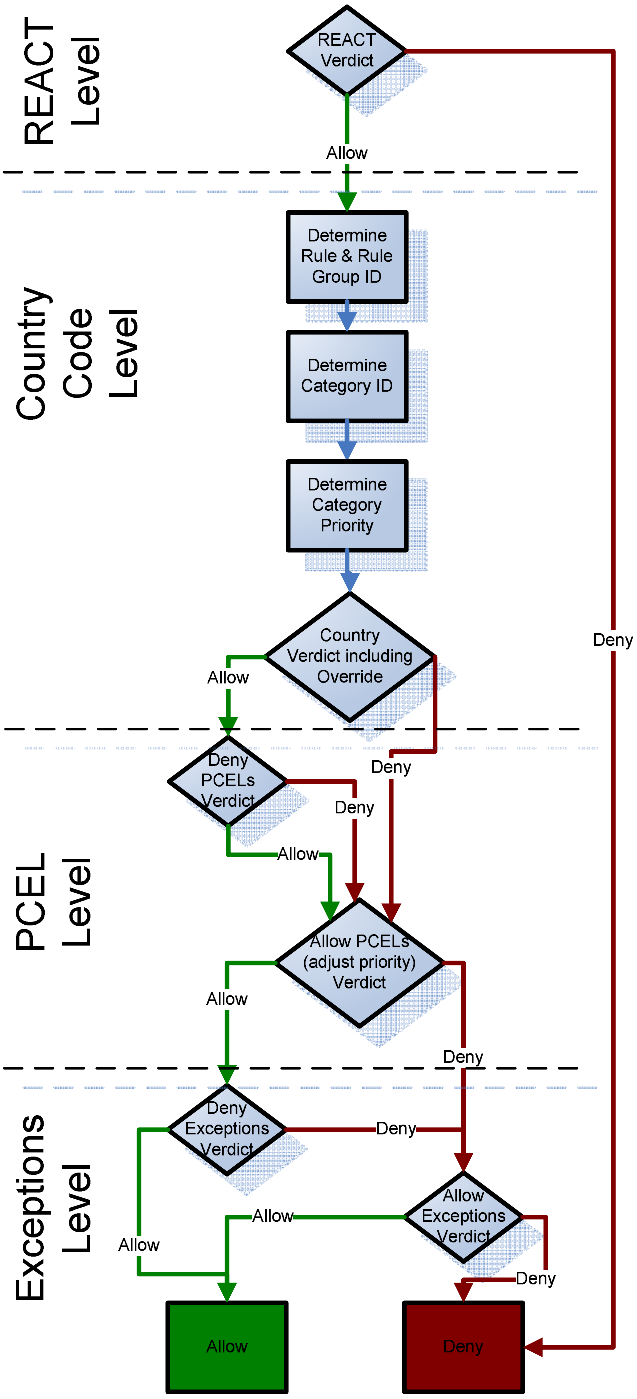

Figure 8-1: Information Flow

The Auto Update Server sends updated IP Addresses to the TOE on a daily basis. The TOE then runs several iterations of checks to determine whether or not the IP address and/or Category (Country) Code will be allowed. As shown in Figure 8-1, the Rule

Group ID, Category Code and IP address are used to determine if a packet is allowed or denied by the TOE. Through its internal components, the TOE processes the flow as shown in Figure 8-1 in order to make the allow or deny decision. If a packet is initially allowed, the TOE checks for overrides, PCELs, and exceptions to see if the packet is possibly denied further down the flow. Conversely, if the packet is initially denied, the TOE checks for overrides, PCELs, and exceptions to see if the packet is possibly allowed further down the flow.

8.4.2.1 REACT Messages

The TOE will be able to receive messages from trusted external entities (REACT Servers) and then block the IP addresses in these messages. The message will also include an amount of time for which to block the IP address. If a specific IP address is specified by the REACT Server multiple times, the total duration of the block will be the sum of the times in each message.

8.4.3 **Quotas**

Quotas for TOE data on transport-layer connections can only be determined by the Security Administrator. If the quota has been reached, all packets above and beyond the quota will be dropped. Quotas can also be placed on controlled connection-oriented resources by the Security Administrator. If the quota has been reached for these resources, the packets will be dropped.

8.4.4 **Revocation of Security Attributes**

The TOE has the ability to restrict revocation of security attributes associated with an Administrator's information flow policy ruleset and services available to unauthenticated Administrators under the control of the TOE which is governed by the Security Administrator. The TOE enforces the following rules with regards to revocation of security attributes:

- Revocation of an Administrator's role (all administrators)
- Revocation of an Administrator's source restrictions
- Changes to the information flow policy ruleset when applied
- Disabling if a service (ICMP Ping) available to unauthenticated Administrators

8.5 Security Management

8.5.1 Roles

The TOE has four default roles that are assigned by the TOE – Security Administrator, Audit Administrator, Cryptographic Administrator (authorized to perform cryptographic initialization and management functions), and Read-Only. It is the TOE's responsibility to ensure that the following conditions are satisfied:

- All administrators shall be able to administer the TOE remotely via the Web-based GUI;
- all three Administrator roles are distinct; that is, there shall be no overlap of operations performed by each default role, with the following exceptions:
    - All roles, including Read-Only, can review the audit trail;
    - The three administrator roles can invoke the self-tests and
    - The three administrator roles can accept alarms/acknowledgements

Below are listed the attributes each Administrator has within the TOE:

8.5.1.1 Security Administrator

The Security Administrator is authorized to perform the following functions on the TOE:

- The Administration section of the TOE provides utilities to manage Administrators, assign Administrators to Roles, set ping access
- Enable and disable security alarms
- Determine the behavior of, disable, enable, modify the behavior of the Audit Trail threshold
- Define policies for accessing the TOE from remote locations
- Create, edit, and remove static ARP table entries for IPv4 and IPv6.
- Unlock the accounts using the Users utility, or the locked out Administrators can try again after the "Minutes Until Locked Account is Unlocked" time passes.
- Create policies, rule groups, alerts, throttles, exception lists, PCELs, manual REACT entries
- Modify the behavior of the Information Flow Policy Rule
- Modify the behavior of the unlocking of locked accounts method
- Modify the behavior of the Password Policy
- Enable/disable ICMP (ping)
- Modify the behavior of the authentication method
- Modify the behavior of the TSF Self-Tests (Periodic Interval)
- Modify the behavior of quotas (i.e. transport layer connections, controlled connection-oriented resources)
- Ability to drop all packets that are above the quota
- Modify the behavior of banners
- Modify the behavior of the Admin Session Policy
- Modify the behavior of Administrators, roles, and categories

8.5.1.2 Audit Administrator

The Audit Administrator is authorized to perform the following functions on the TOE:

- Delete audit records by specifying a percentage of the audit records to remove. The oldest records will be deleted.

8.5.1.3 Cryptographic Administrator

The Cryptographic Administrator is authorized to perform the following functions on the TOE:

- Install and update the x509 certificate used by the server by either:
    1. Generating a new certificate request and private key on the TOE and then uploading the signed certificate to the TOE
    2. Uploading an PKCS12 file with certificate and private key
- Install/update the x509 certificate required for client certificate authentication
- Enable/disable the client certificate requirement
- Configure the IPsec tunnel settings
- Set HTTPS access and manage server and client certificates.

8.5.1.4 All Administrators

All Administrators are authorized to perform the following functions on the TOE:

- Modify the behavior of the security alarm acknowledgement
- Modify the behavior of TSF Self-Tests (Perform)

In addition to the Administrators aforementioned, authorized IT entities also have privileges within the TOE – the NTP Server can modify the behavior of timestamps and the Auto-Update Server can modify the behavior of the category databases. The REACT Server may alter the Flow Control Policy. The RMC Server may execute commands in place of the Remote Administrator.

The TOE also allows for importing and exporting. The TOE allows the running configuration to exported to a file. When multiple TOEs are used within a network, the configuration can be configured on one TOE, exported, and then loaded onto the other TOEs.

8.5.2 **Access Control Mechanisms**

The TOE contains several access control mechanisms implemented to manage the login process. Listed below are those access control mechanisms:

- HTTP Access Restrictions
    - HTTPS (HTTP over SSL) controls access for the graphical user interface (GUI). This utility allows administrators to change the port on which the HTTPS server listens (port 443 by default). IPv4 and IPv6 addresses and

ranges define the source addresses from which Administrators may connect to the PoliWall web interface.

- User IP Address Access Restrictions
    - IPv4 and IPv6 addresses may be specified for each user to identify which sources a specified Administrator may connect from

- IP Address Restrictions
    - The Administrative interface is used to assign the IPv4 and/or IPv6 address to the PoliWall administration interface. Only one IPv4 and one IPv6 address can be entered.
    - The following information is needed if using IPv4:
        - IPv4 Address
        - IPv4 Gateway
        - MTU
    - The following information is needed if using IPv6:
        - IPv6 Addressing Mode (e.g. Static or Stateless Autoconfig)
        - IPv6 Address
        - IPv6 Link Local Address (fixed)
        - IPv6 Gateway
        - MTU

After a user's session is established, it may be terminated due to inactivity timeout, due to maximum session duration timeout, by the session being terminated by another user, or by the user being disabled by another user.

The table below shows the eleven main menu-driven configuration options and their subcategories.

Main Menu Options **(listed on the left side of the user interface)**	**Sub-Categories** **(drop-down menus)**
Rules	Rule Groups
Policies	Policies Exception Lists PCELs
REACT	Configuration Manually Blocked IPs Auto Blocked IPs
Live Stats	Stats for Rule Groups
Logs	IPv4 Packet Logs IPv6 Packet Logs System Logs External Syslog Servers Purge Logs
Users	Accounts Maintenance Users Change Password
Network	Admin Interface Bridging Interface IPsec Settings

	Arp Table
Configuration	General Settings Bypass Settings Cryptographic Settings Alarm Settings Banner Settings RMC Settings HTTP Settings SNMP Settings NTP Settings HIPPIE Providers PCEL Providers
Update	Software
System	Active Sessions Reboot Shutdown Self-Test Maintenance Mode System Information Import/Export
Logout	Logout

Table 8-2: Menu Options for the TOE

8.5.3 Security Attributes

The TOE has the ability to enforce the unauthenticated information flow SFP to restrict the ability to change, default, query, and modify the security attributes by the Security Administrator. The Security Administrator has the privileges to create attributes to administrators that are added to the TOE. These attributes can be used in specifying the information flow policy rules. As an aside, the attributes associated with stateful packet inspection are not expected to be managed by the Security Administrator.

The unauthenticated information flow SFP must provide default values for security attributes that are used to enforce the SFP. The attributes define the default information flow policy ruleset, which is “deny all” network traffic. Alternative initial values must be specified in order to override the default values when an object or information is created.

8.5.4 Memory Management

Memory management takes place at both the kernel level and user-space program levels. In the kernel, structures are zeroed out upon receipt of the allocation and immediately before return to free memory via deallocation. In the user-space programs, the destructors of all classes will “zeroize” the memory they used before exiting the destructor.

8.6 Protection of the TSF

The TOE comes pre-installed with a self-signed SSL certificate issued to TechGuard Labs. The SSL server certificate is used to establish a secure encrypted session to the PoliWall configuration application. The appliance includes a generic server certificate. The pre-installed certificate will be overwritten after successfully configuring and installing a new server certificate.

The PoliWall client CA certificate specifies the certificate authority required to issue client certificates which identify Administrators connecting to the PoliWall. When this feature is enabled by the Security Administrator, the internal web server will accept a client certificate that is installed on the Administrator's system. However, it is not required to gain access to the PoliWall administration GUI. Only Administrators with a valid client certificate will be able to access the logon screen. If the Client CA Certificate expires, causing a lockout condition, the Maintenance Mode is used to resend the certificate.

HTTPS (HTTP over SSL) controls access for the graphical user interface (GUI). This utility allows administrators to change the port on which the HTTPS server listens (port 443 by default). IPv4 and IPv6 addresses and ranges define the source addresses from which Administrators may connect to the PoliWall web interface.

Additionally, the TOE utilizes IPsec (Internet Protocol Security) which has two modes of operation – Transport Mode and Tunnel Mode.

Transport Mode **-** In transport mode only the payload (message) of the IP packet is encrypted. The routing is intact since the IP header is neither modified nor encrypted; however, when the Authentication Header is used, the IP addresses cannot be translated through NAT, as this will invalidate the hash value. The transport and application layers are always secured by hash so they cannot be modified in any way (for example by translating the port numbers). Transport mode is used for host-to-host communications.

Tunnel Mode - In tunnel mode, the entire IP packet is encrypted. It must then be encapsulated into a new IP packet for routing to work. Tunnel mode is used for network-to-network, host-to-network and host-to-host communications over the Internet.

8.7 Self Protection (ADV_ARC.1)

The TOE will maintain a secure state even when failures to the Auto Update, PoliWall process, msglogd, syslogd, pktlogd, and pktlog6d occur. There is an internal component that continuously polls these processes and restarts them if they have failed. The TOE will also maintain and provide reliable timestamps to Administrators. In order to maintain the integrity of the TOE, the TSF will run a suite of self-tests during initial start-up, periodically during normal operation, and at the request of the authorized Administrator in order to demonstrate the correct operation of the TOE. These tests can be run at predefined times set by the Security Administrator, or they can be manually run by authorized administrators from the Remote Management Console. There is an internal component that performs a hash of the data and executable code and compares these hashes to those stored in the hash database to ensure that the data and code has not been modified. All authorized Administrators will be able to verify the integrity of TOE data and stored TOE executable code.

The TOE maintains individual sessions associated with Administrators once they authenticate. The TSF maintains the Administrator's identification (i.e. username/password) as part of a session to prevent interference between Administrator actions. An Administrator's access to the TOE and TOE data is also determined upon session establishment by being associated with a role which has specific functions that can be performed. The only function an unauthenticated Administrator is allowed to perform on the TOE is ICMP; however, that is only if the Security Administrator has enabled this function.

The TOE has the ability to restart certain processes if they have failed. For instance, if the AutoUpdate server has failed, there is an internal component of the TOE that will restart it. Similarly, if the PoliWall process fails, it will be restarted. If the audit functions go down, the TOE holds the audit information in queue until they are restarted and it's able to write to the audit trail.

The TOE also provides encryption of TOE data and a secure communication path between the remote Administrators and the TOE.

8.8 Trusted Path

The TOE will provide an encrypted communication channel between itself and another trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure. The TOE can initiate communication via this trusted channel for updates of the system time, category code database, PCEL database, REACT messages, SNMP Traps, and RMC connections.

The TOE must ensure that a secure communication path between itself and remote Administrators that is distinct from other communication paths. The TOE must also provide Administrators with an assured identification of its end points and protect communicated data from be modified and/or disclosed. Remote Administrators must be able to initiate communication to this trusted line of communication. Before this occurs, however, proper authentication is required by the Administrator to access this trusted path. This trusted path is accomplished through HTTP over SSL encryption. For more information on the trusted path used by the TOE, refer to section 8.6 Protection of the TSF.

8.9 Resource Utilization

In the event of the failures of the Auto Update module, PoliWall process module (remote administration functions and access control), and auditing modules (msglogd, syslogd, pktlogd, pktlog6d) the TOE will maintain and operate in a secure state until these failures have come back online. Information flow control will remain in operation during this time.

Through the Quality of Service (QoS) policy, the TOE allows the Administrator to give traffic from specified countries a higher priority than traffic from other countries. When the total amount of traffic reaches the configured bandwidth limit, traffic from the high QoS countries will be allowed through the PoliWall before traffic from other countries. Countries in an active Throttle will not be given high QoS even if they are selected here. Quality of Service can be configured using either the SVG map or a list box.

If the PoliWall has had a configuration fault, it will enter Maintenance Mode. In Maintenance Mode, there are limited recovery options that may be performed. The following are options that Administrators have if the PoliWall becomes unstable due to a configuration fault:

- Enter Maintenance Mode - This will allow the Administrator to bring the PoliWall into Maintenance Mode to enable the rest of the options. Any of the three default roles may perform this action.
- View Alarms - View the Alarms that were raised on the TOE to cause it to enter Maintenance Mode. Any of the three default roles may perform this action.
- Key Zeroization - Clear the Cryptographic Keys for use with the SSL Web Server. Only the Cryptographic Administrator may perform this action.

- Purge IPv4 Packet Log Message - Delete a percentage of the messages from the IPv4 Packet Log. Only the Audit Administrator may perform this action.
- Purge IPv6 Packet Log Message - Delete a percentage of the messages from the IPv6 Packet Log. Only the Audit Administrator may perform this action.
- Purge System Log Message - Delete a percentage of the messages from the System Log. Only the Audit Administrator may perform this action.
- View System Log - View the log records in the System Log. Only the Audit Administrator may perform this action.
- Reset Admin Account - Reset the username, password, roles, and login restrictions for the default admin account. Only the Security Administrator may perform this action.
- Reset Admin Interface - Reset the IP address and HTTPS restrictions for the Administrative Interface. Only the Security Administrator may perform this action.
- Reset Configuration - Reset the entire Configuration of the TOE to the default from the factory. Only the Security Administrator may perform this action.

Additionally, the TOE can enforce maximum quotas on transport layer representation and controlled connection-oriented resources that subjects can use simultaneously.

8.10 TOE Access

8.10.1 Access Restrictions

Access to the TOE is controlled by the Administrator's IP address. After a given amount of time (determined by the Security Administrator), the interactive session will be terminated due to inactivity. Before a session is established, the TOE will display an advisory banner warning against unauthorized use of the TOE. The TOE can deny a session being started based on IP address, time, and day.

8.10.2 Logon Restrictions

The relationship between HTTPS access restrictions and user account IP restrictions for the logon process is:

- If HTTPS access restrictions have been configured, the IP address of the connecting machine is evaluated before presenting the logon dialog.
- If user account IP restrictions are in effect, they are evaluated before permitting access to the PoliWall administration GUI.

9 TOE Summary Specification Rationale

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

Security Function	Security Functional Components
Security Audit	FAU_ARP.1 Security Alarms
	FAU_ARP_EXT.1 Security Alarm Acknowledgement
	FAU_GEN.1 Audit Data Generation
	FAU_GEN.2 User Identity Association
	FAU_SAA.1 Potential Violation Analysis
	FAU_SAR.1 Audit Review
	FAU_SAR.2 Restricted Audit Review
	FAU_SAR.3 Selectable Audit Review
	FAU_STG.1 Protected Audit Trail Storage
	FAU_STG.3 Action In Case Of Possible Audit Data Loss
	FAU_STG.4 Prevention of Audit Data Loss
Cryptographic Support	FCS_CKM.1 Cryptographic Key Generation
	FCS_CKM.4 Cryptographic Key Destruction
	FCS_COP.1(1) Cryptographic Operation
	FCS_COP.1(2) Cryptographic Operation
User Data Protection	FDP_IFC.1 Subset Information Flow Control
	FDP_IFF.1 Simple Security Attributes
	FDP_RIP.1 (1) Subset Residual Information Protection

Security Function	Security Functional Components
	FDP_RIP.1 (2) Subset Residual Information Protection
Identification and Authentication	FIA_AFL.1 Authentication Failure Handling
	FIA_ATD.1 User Attribute Definition
	FIA_SOS.2 TSF Generation of Secrets
	FIA_UAU.1 Timing of Authentication
	FIA_UAU.5 Multiple Authentication Mechanisms
	FIA_UID.2 User Identification Before Any Action
	FIA_USB.1 User-Subject Binding
Security Management	FMT_MOF.1 Management of Security Functions Behavior
	FMT_MSA.1 Management of Security Attributes
	FMT_MSA.3 Static Attribute Initialization
	FMT_MTD.1 Management of TSF Data
	FMT_MTD.2 Management of limits on TSF Data
	FMT_REV.1 Revocation
	FMT_SMF.1 Specification of management functions
	FMT_SMR.2 Restrictions on Security Roles
Protection of the TSF	FPT_FLS.1 Failure of preservation of secure state
	FPT_STM.1 Reliable time stamps
	FPT_TST.1 TSF testing
Resource Utilization	FRU_FLT.1(1) Degraded Fault Tolerance
	FRU_FLT.1(2) Degraded Fault Tolerance
	FRU_FLT.1(3) Degraded Fault Tolerance

Security Function	Security Functional Components
	FRU_FLT.2 Limited Fault Tolerance
	FRU_PRS.1 Limited Priority of Service
	FRU_RSA.1 Maximum Quotas
TOE Access	FTA_SSL.3 TSF-Initiated Termination
	FTA_TAB.1 Default TOE Access Banners
	FTA_TSE.1 TOE Session Establishment
Trusted Path/Channels	FTP_ITC.1 Inter-TSF Trusted Channel
	FTP_TRP.1 Trusted Path

Table 9-1: Security Functional Components

9.1.1 Security Audit

Section 1.3.4.1 states that the TOE is able to generate security alarms. This is described in detail in section 8.1.2 where it states the TOE is able to generate security alarm messages that identify potential security violations. The message is displayed at the remote console if an administrator is already logged in, or when an administrator logs in if the alarm message has not been acknowledged. In addition, the TOE provides an audible alarm that can be configured to sound an alarm if desired by the Security Administrator.

Section 1.3.4.1 states that the TOE has the ability to produce audit logs. This is described in detail in section 8.1.1 where it states the TOE will generates audit reports for the start-up and shutdown of the audit functions and for all events listed in Table 6-2 Auditable Events. Each audit record captures the date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event. Each event is associated with the user that caused the event. Tables 8-1, 8-2, and 8-3 display what an administrator sees when at the audit log interface. For failed login attempts no user association is required because the user is not under TSF control until after a successful identification/authentication. For these requirements, user refers to the userid for authorized users, and a network identifier for unauthenticated network traffic.

The following rules apply to the audited events that are known to indicate a potential security violation: Security Administrator specified number of authentication failures, any failure of the TSF self-tests, failure to automatically update the country code DB and meet the threshold for audit log. Once this threshold has been met, an alarm is generated.

Section 1.3.4.1 briefly describes the rules administrators have that pertain to the audit logs. Section 8.1.2 describes in detail the overall functionality that administrators can perform with said logs. All Administrators are authorized to read, search, and sort the audit data; however, only the Audit Administrator is authorized to delete the audit data. Administrators can perform searches and sorting of the audit data based on user identity, command type (for user actions) and date and time (for flow control decisions).

All audit records are protected from unauthorized deletion and all unauthorized modifications are prevented by the TOE. Additionally, if the storage capacity for the audit trail meets the threshold previously set by the Security Administrator, an alarm is generated and the Security Administrator is allowed to overwrite the oldest stored audit records.

Based on the above information, the TOE enforces the FAU_ARP.1, FAU_ARP_EXT.1, FAU_GEN.1, FAU_GEN.2, FAU_SAA.1, FAU_SAR.1, FAU_SAR.2, FAU_SAR.3, FAU_STG.1, FAU_STG.3 and FAU_STG.4 requirements as stated in section 6.

9.1.2 **Cryptographic Support**

Section 1.3.4.2 displays a table that provides how the TOE intends to provide encryption. Section 8.2 provides greater detail for those encryption mechanisms to be used. The TOE uses AES with 256 bit keys for key generation, RSA with 2048 bit keys for encryption and decryption and SHA-1 with 160 bit keys and SHA-256 with 256 bit keys for cryptographic hashing services. Additionally, TLS v1.0 is used for secure communication between remote administrators and the TOE.

Based on the above information, the TOE enforces the FCS_CKM.1, FCS_CKM.4 FCS_COP.1(1) and FCS_COP.1(2) requirements as stated in section 6.

9.1.3 **User Data Protection**

The unauthenticated information flow SFP is enforced on the source subject, destination subject, network packets and pass information. Section 1.3.4.5 provides general guidance in regards to the flow of information. Section 8.4 provides greater and states that the information flow is explicitly denied if the presumed source ID of the information received by the TOE is not included in the set of source identifiers for the source subject. This flow is based upon the port, protocol, and IP address. Whenever a packet is received that is not associated with an allowed established session (e.g., the SYN flag is set without the ACK flag being set), the information flow policy ruleset is applied to the packet; Otherwise, the TSF associates a packet with an allowed established session.

Section 8.4.1 provides more detail with how users are associated with roles. It illustrates that the information that associates a user to a role after authentication is stored in the session object that is created for each authenticated session in the PoliWall process. This object contains the current user and the current role for the session. The session identifier is passed in with the XML-messages for the function calls. Additionally, PHP also stores tracks sessions by setting a cookie on the client. When the web browser presents the cookie, PHP can identify its session that way and have the session id for the XML-message stored internally to be passed in.

Based on the above information, the TOE enforces the FDP_IFC.1, FDP_IFF.1, FDP_RIP.1(1) and FDP_RIP.1(2) requirements as stated in section 6.

9.1.4 **Identification and Authentication**

Section 1.3.4.3 speaks about the TOE's password policy. Section 8.3 goes into greater detail about the TOE's authentication process and the rules that govern that process. The TOE is able to provide a mechanism to generate passwords that meet a predefined ruleset for authentication and access control. When 2-25 unsuccessful authentication attempts occur for Administrators attempting to authenticate remotely, the Security Administrator can prevent the session from occurring until the Administrator's password is reset or until a Security Administrator defined time period has elapsed**.** Administrators are only allowed to perform ICMP as unauthenticated users, only if the Security Administrator has enabled this function. Otherwise, all Administrators must be identified and authenticated prior to performing any actions on the TOE.

The TOE maintains the following attributes for Administrators: username, password, certificate, security descriptor, role and IP address. All Administrators are associated with their respective attributes on the TOE.

Based on the above information, the TOE enforces the FIA_AFL.1, FIA_ATD.1, FIA_SOS.2, FIA_UAU.1, FIA_UAU.5, FIA_UID.2 and FIA_USB.1 requirements as stated in section 6.

9.1.5 **Security Management**

Section 1.3.4.4 states that there are specific administrators of the TOE and they have certain abilities. Section 8.5.1 goes into great detail about the administrators of the TOE as well as their attributes. Administrators are authorized to perform the functions of the TOE as specified in Tables 6-3 Management Functions of the TOE and 6-4 Management of TSF Data. The Security Administrator is authorized to change default, query, modify and provide restrictive default values for the security attributes referenced in the indicated policies that enforce the unauthenticated information flow SFP. Only the Security Administrator is authorized to specify alternative initial values to override the default values when an object or information is created. The chart in Section 8.4.2 displays how information is permitted (or not) to flow through the TOE.

Additionally, the Security Administrators are authorized to specify the limits for quotas on transport-layer connections and controlled connection-oriented resources based on IP address and Category code. If the quotas meet or exceed the limits set by the Security Administrator, the TOE drops all packets above the quota.

Security Administrators also have the ability to revoke security attributes associated with Administrators, information flow policy ruleset and services available to unauthenticated Administrators.

In addition to the Security Administrator, the TOE also maintains the roles for Cryptographic Administrators, Audit Administrators, and read-only. All administrative roles are able to manage the TOE remotely and do not overlap.

Based on the above information, the TOE enforces the FMT_MOF.1, FMT_MSA.1, FMT_MSA.3, FMT_MTD.1, FMT_MTD.2, FMT_REV.1, FMT_SMF.1, and FMT_SMR.2 requirements as stated in section 6.

9.1.6 **Protection of the TSF**

Section 1.3.4.9 states that the TOE is able to provide a secure state when certain processes/modules fail. Section 8.6 describes in detail what process and modules may fail, and if they do, how is that failure mediated. The TOE is able to restart the following modules if they go down: Auto Update, PoliWall Process, msglogd, syslogd, pktlogd, and pktlog6d. This allows for the TOE to preserve a secure state.

Section 8.6 also states that the TOE provides reliable time stamps for audit records. The timestamps are maintained and verified through the Network Time Protocol (NTP) Server.

The TOE performs self tests during initial start-up, periodically during normal operation and at the request of the authorized Administrator to demonstrate the correct operation of the TSF.

Based on the above information, the TOE enforces the FPT_FLS.1, FPT_STM.1 and FPT_TST.1 requirements as stated in section 6.

9.1.7 **Resource Utilization**

Section 1.3.4.7 briefly speaks to how the TOE mitigates failures to its modules. Section 8.8 goes into detail describing what happens when failures to the TOE occur. When the Auto Update module goes down, the TOE ensures the operation of information flow control. When the PoliWall Process module goes down, the TOE ensures the operation of remote administration functions and access control. Similarly, when any of the auditing modules go down, the TOE ensures the operation of the audit functions.

Section 1.3.4.7 briefly describes how flows are permitted or denied. Section 8.8 describes in detail the Quality of Service policy which allows for information to flow on a priority basis. It states the unauthenticated flow control gains access to the TOE based on a priority that is set by the Security Administrator. Maximum quotas on bandwidth can be set by Security Administrators for transport layer representation and controlled connection-oriented resources that subjects can use simultaneously. Once the network traffic for a particular category code exceeds the quota all packets which exceed that quota will be dropped.

Based on the above information, the TOE enforces the FRU_FLT.1(1), FRU_FLT.1(2), FRU_FLT.1(3), FRU_FLT.2, FRU_PRS.1 and FRU_RSA.1 requirements as stated in section 6.

9.1.8 **TOE Access**

Section 1.3.4.8 gives a brief description on how administrators gain access to the TOE and what occurs when they attempt to gain access. Section 8.9 speaks specifically to access (whether restricted or granted) to the TOE. Session establishment can be denied based on the Admin Session Policy, which refers to the source restriction (IP address), time and day. Once an administrator's remote session has been inactive for a predefined set of time set by the Security Administrator, the session will be terminated.

Prior to establishing a session, the TOE displays an advisory warning message regarding unauthorized use of the TOE. The access banner applies whenever the TOE provides a prompt for identification and authentication (e.g., administrators). This is to advise Administrators of warnings regarding the unauthorized use of the TOE and to provide the Security Administrator with control over what is displayed (e.g., if the Security Administrator chooses, they can remove banner information that informs the Administrator of the product and version number).

Based on the above information, the TOE enforces the FTA_SSL.3, FTA_TAB.1 and FTA_TSE.1 requirements as stated in section 6.

9.1.9 **Trusted Path/Channels**

Section 1.3.4.6 speaks about the trusted communication path for the administrators to gain access to the TOE. Section 8.7 specifically describes how a secure communication path is provided to the TOE. The TOE provides the trusted channel to initiate communication in order to update system time, to update the Category Code Database, to communicate with the SNMP server, to communicate with the REACT server, and to communicate with the RMC Server.

The TOE provides the trusted path for initial Administrator authentication and all administrative actions that occur remotely.

Based on the above information, the TOE enforces the FTP_ITC.1 and FTP_TRP1 requirements as stated in section 6.

10 Security Problem Definition Rationale

10.1 Security Objectives Rationale

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

Assumption	Objective	Rationale
A.PHYSICAL Users responsible for management of the operational environment exercise due diligence to update the TOE with the latest patches and patch the Operational Environment (e.g., OS and database) so they are not susceptible to network attacks.	OE.PHYSICAL Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the IT environment.	OE.PHYSICAL maps to A.PHYSICAL to ensure that the TOE is located is updated with the latest patches.
A.NO_TOE_BYPASS Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the environment.	OE.NO_TOE_BYPASS Information cannot flow between external and internal networks located in different enclaves without passing through the TOE.	OE.NO_TOE_BYPASS maps to A.NO_TOE_BYPASS to ensure that information from the internal networks cannot flow directly to the external networks without first passing through the TOE.

Table 10-1: Assumption to Objective Mapping

Threat/Policy	**Objective**	**Rationale**

T.ADDRESS_MASQUERADE A user on one interface may masquerade as a user on another interface to circumvent the TOE policy.	O.MEDIATE The TOE must mediate the flow of information between sets of TOE network interfaces or between a network interface and the TOE itself in accordance with its security policy.	O.MEDIATE (FDP_IFF.1, FDP_IFC.1, FMT_REV.1, ADV_ARC.1) counters this threat by ensuring that all network packets that flow through the TOE are subject to the information flow policies. The rules in each of the policies ensure that the network identifier in a network packet is in the set of network identifiers associated with a TOE's network interface. Therefore, if a user supplied a network identifier in a packet that was associated with a TOE network interface other than the one the user supplied the packet on, the packet would not be allowed to flow through the TOE, or access TOE services. This would, for example, prevent a user from sending a packet from the Internet claiming to be on a machine on the protected enclave.
T.ADMIN_ERROR An administrator may incorrectly install or configure the TOE, or install a corrupted TOE resulting in ineffective security mechanisms.	O.ROBUST_ADMIN_GUIDANCE The TOE will provide administrators with the necessary information for secure delivery and management.	O.ROBUST_ADMIN_GUIDANCE (ALC_DEL.1, AGD_PRE.1, AGD_OPE.1) help to mitigate this threat by ensuring the TOE administrators have guidance that instructs them how to administer the TOE in a secure manner and to provide the administrator with instructions to ensure the TOE was not corrupted during the delivery process. Having this guidance helps to reduce the mistakes that an administrator might make that could cause the TOE to be configured in a way that is unsecure.

	O.ADMIN_ROLE The TOE will provide an administrator role to isolate administrative actions.	O.ADMIN_ROLE (FMT_SMR.2) plays a role in mitigating this threat by limiting the functions an administrator can perform in a given role. So for example, the Audit Administrator could not make a configuration mistake that would impact the TOE's ability to mediate information flow.
	O.MANAGE The TOE will provide all the functions and facilities necessary to support the administrators in their management of the security of the TOE, and restrict these functions and facilities from unauthorized use.	O.MANAGE (FMT_MTD.1, FMT_MTD.2, FMT_MSA.1, FMT_MSA.3, FMT_MOF.1, FMT_SMF.1, FAU_SAR.1, FAU_SAR.2, FAU_SAR.3, FAU_STG.1, FAU_STG.3, FAU_STG.4, FAU_ARP_EXT.1) also contributes to mitigating this threat by providing administrators the capability to view configuration settings. For example, if the Security Administrator made a mistake when configuring the ruleset, providing them the capability to view the rules affords them the ability to review the rules and discover any mistakes that might have been made.
T.ADMIN_ROGUE An administrator's intentions may become malicious resulting in user or TSF data being compromised.	O.ADMIN_ROLE The TOE will provide an administrator role to isolate administrative actions.	O.ADMIN_ROLE (FMT_SMR.2) mitigates this threat to a limited degree by limiting the functions available to an administrator. This is somewhat different than the part this objective plays in countering T.ADMIN_ERROR, in that this presumes that separate individuals will be assigned separate roles. If the Audit Administrator's intentions become malicious they would not be able to render the TOE unable to enforce its information flow policies. On the other hand, if the Security Administrator becomes malicious they could affect the information flow policy, but the Audit Administrator may be able to detect those actions.

T.AUDIT_COMPROMISE A malicious user or process may view audit records, cause audit records to be lost or modified, or prevent future audit records from being recorded, thus masking a user’s action.	O.AUDIT_PROTECTION The TOE will provide the capability to protect audit information.	O.AUDIT_PROTECTION (FAU.SAR.2, FAU_STG.1, FAU_STG.3, FAU_STG.4, FMT_MOF.1) contributes to mitigating this threat by controlling access to the audit trail. No one is allowed to modify audit records, the Audit Administrator is the only one allowed to delete the audit trail. The TOE has the capability to prevent auditable actions from occurring if the audit trail is full.
	O.RESIDUAL_INFORMATION The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated.	O.RESIDUAL_INFORMATION (FDP.RIP.1, FCS_CKM.4) prevents a user not authorized to read the audit trail from access to audit information that might otherwise be persistent in a TOE resource (e.g., memory). By ensuring the TOE prevents residual information in a resource, audit information will not become available to any user or process except those explicitly authorized for that data.
	O.SELF_PROTECTION The TSF will maintain a domain for its own execution that protects itself and its resources from external interference, tampering, or unauthorized disclosure.	O.SELF_PROTECTION (ADV_ARC.1, FTP_ITC.1, FTP_TRP.1, FPT_FLS.1, FRU_FLT.1(1), FRU_FLT.1(2), FRU_FLT.1(3, FRU_FLT.2) contributes to countering this threat by ensuring that the TSF can protect itself from users. ADV_ARC.1 provides the security architecture description of the security domains maintained by the TSF that are consistent with the SFRs. Since self-protection is a property of the TSF that is achieved through the design of the TOE and TSF, and enforced by the correct implementation of that design, self-protection will be achieved by that design and implementation.

T.CRYPTO_COMPROMISE A malicious user or process may cause key, data, or executable code associated with the cryptographic functionality to be inappropriately accessed (viewed, modified, or deleted), thus compromise the cryptographic mechanism and the data protected by those mechanisms.	OE.CRYPTANALYTIC Cryptographic methods used in the IT environment shall be interoperable with the TOE, and should be resistant to cryptanalytic attacks (i.e., will be of adequate strength to protect unclassified Mission Support, Administrative, or Mission Critical data).	OE. CRYPTANALYTIC (FPT_ITC.1 and FTP_TRP.1) ensures that encryption is used on the communication channel between authorized IT entities and the TOE and that an administrator can be assured that they are communicating with the TOE.
T.MASQUERADE An unauthenticated user may masquerade as an authorized user or an authorized IT entity to gain access to data or TOE resources.	O.ROBUST_TOE_ACCESS The TOE will provide mechanisms that control a user's logical access to the TOE and to explicitly deny access to specific users when appropriate	O.ROBUST_TOE_ACCESS (FTA_TSE.1, FIA_UID.2, FIA_SOS.2, FTA_SSL.3, AVA_VAN.3, FIA_AFL.1, FIA_ATD.1, FIA_UAU.1, FIA_UAU.5) mitigates this threat by controlling the logical access to the TOE and its resources. By constraining how and when authorized users can access the TOE, and by mandating the type and strength of the authentication mechanism this objective helps mitigate the possibility of a user attempting to login and masquerade as an authorized user. In addition, this objective provides the administrator the means to control the number of failed login attempts a user can generate before an account is locked out, further reducing the possibility of a user gaining unauthorized access to the TOE. Additionally this mechanism prevents unauthenticated users from accessing any of the TOE's configuration information or altering the TOE's configuration in any way.

	O.TRUSTED_PATH The TOE will provide a means to ensure users are not communicating with some other entity pretending to be the TOE, and that the TOE is communicating with an authorized IT entity and not some other entity pretending to be an authorized IT entity.	O.TRUSTED_PATH (FTP_TRP.1, FTP_ITC.1) ensures that the communication path end points between the TOE and authorized users (remote administrators, authorized IT entities) are defined. This mechanism allows the TOE to be assured that it is communicating with an authorized user. This also ensures that the transmitted data cannot be disclosed (e.g., encrypted). The protection offered by this objective is limited to TSF data and security attributes (e.g., proxy user's user data is not protected, since their session communication does not require encryption or any other form of protection).
T.FLAWED_DESIGN Unintentional or intentional errors in requirements specification or design of the TOE may occur, leading to flaws that may be exploited by a malicious user or program.	O.CHANGE_MANAGEMENT The configuration of, and all changes to, the TOE and its development evidence will be analyzed, tracked, and controlled throughout the TOE's development.	O.CHANGE_MANAGEMENT (ALC_CMC.4, ALC_CMS.4, ALC_DVS.1, ALC_FLR.2, ALC_LCD.1) plays a role in countering this threat by requiring the developer to provide control of the changes made to the TOE's design. This includes controlling physical access to the TOE's development area, and having an automated configuration management system that ensures changes made to the TOE go through an approval process and only those persons that are authorized can make changes to the TOE's design and its documentation.

	O.SOUND_DESIGN The design of the TOE will be the result of sound design principles and techniques; the design of the TOE, as well as the design principles and techniques, are adequately and accurately documented.	O.SOUND_DESIGN (ADV_FSP.4, ADV_TDS.3) counters this threat, to a degree, by requiring that the TOE be developed using sound engineering principles. By accurately and completely documenting the design of the security mechanisms in the TOE, including a security model, the design of the TOE can be better understood, which increases the chances that design errors will be discovered.
	O.VULNERABILITY_ANALYSIS_ TEST The TOE will undergo appropriate independent vulnerability analysis and penetration testing to demonstrate the design and implementation of the TOE does not allow attackers with medium attack potential to violate the TOE's security policies.	O.VULNERABILITY_ANALYSIS_TEST (AVA_VAN.3, ADV_ARC.1, ADV_FSP.4, ADV_TDS.3) requires that the TOE be developed using sound engineering principles. By accurately and completely documenting the design of the security mechanisms in the TOE, including a security model, the design of the TOE can be better understood, which increases the chances that design errors will be discovered. The design of the TOE is independently analyzed for design flaws. Having an independent party perform the assessment ensures an objective approach is taken and may find errors in the design that would be left undiscovered by developers that have a preconceived incorrect understanding of the TOE's design. Since self-protection is a property of the TSF that is achieved through the design of the TOE and TSF, and enforced by the correct implementation of that design, self-protection will be achieved by that design and implementation.

T.FLAWED_IMPLEMENTATION Unintentional or intentional errors in implementation of the TOE design may occur, leading to flaws that may be exploited by a malicious user or program.	O.CHANGE_MANAGEMENT The configuration of, and all changes to, the TOE and its development evidence will be analyzed, tracked, and controlled throughout the TOE's development.	O.CHANGE_MANAGEMENT (ALC_CMC.4, ALC_CMS.4,ALC_DVS.1, ALC_FLR.2, ALC_LCD.1,) This objective plays a role in mitigating this threat in the same way that the flawed design threat is mitigated. By controlling who has access to the TOE's implementation representation and ensuring that changes to the implementation are analyzed and made in a controlled manner, the threat of intentional or unintentional errors being introduced into the implementation are reduced. In addition to documenting the design so that implementers have a thorough understanding of the design,
	O.SOUND_IMPLEMENTATION The implementation of the TOE will be an accurate instantiation of its design, and is adequately and accurately documented.	O.SOUND_IMPLEMENTATION (ADV_IMP.1, ADV_TDS.3, ALC_TAT.1) requires that the developer's tools and techniques for implementing the design are documented. Having accurate and complete documentation, and having the appropriate tools and procedures in the development process helps reduce the likelihood of unintentional errors being introduced into the implementation. Although the previous three objectives help minimize the introduction of errors into the implementation,
	O.THOROUGH_FUNCTIONAL_ TESTING The TOE will undergo appropriate security functional testing that demonstrates the TSF satisfies the security functional requirements.	O.THOROUGH_FUNCTIONAL_ TESTING (ATE_FUN.1, ATE_COV.2, ATE_DPT.2, ATE_IND.2) increases the likelihood that any errors that do exist in the implementation (with respect to the functional specification, high level, and low-level design) will be discovered through testing.

	O.VULNERABILITY_ANALYSIS_ TEST The TOE will undergo appropriate independent vulnerability analysis and penetration testing to demonstrate the design and implementation of the TOE does not allow attackers with medium attack potential to violate the TOE's security policies.	O.VULNERABILITY_ANALYSIS_TEST (AVA_VAN.3, ADV_ARC.1, ADV_FSP.4, ADV_TDS.3) requires that the TOE be developed using sound engineering principles. By accurately and completely documenting the design of the security mechanisms in the TOE, including a security model, the design of the TOE can be better understood, which increases the chances that design errors will be discovered. The design of the TOE is independently analyzed for design flaws. Having an independent party perform the assessment ensures an objective approach is taken and may find errors in the design that would be left undiscovered by developers that have a preconceived incorrect understanding of the TOE's design. Since self-protection is a property of the TSF that is achieved through the design of the TOE and TSF, and enforced by the correct implementation of that design, self-protection will be achieved by that design and implementation.
T.POOR_TEST Lack of or insufficient tests to demonstrate that all TOE security functions operate correctly (including in a fielded TOE) may result in incorrect TOE behavior being undiscovered.	O.CORRECT_ TSF_OPERATION The TOE will provide the capability to test the TSF to ensure the correct operation of the TSF in its operational environment.	O.CORRECT_ TSF_OPERATION (FPT_TST.1) ensures that once the TOE is installed at a customer's location, the capability exists that the integrity of the TSF (hardware and software) can be demonstrated, and thus providing end users the confidence that the TOE's security policies continue to be enforced.

	O.THOROUGH_FUNCTIONAL_ TESTING The TOE will undergo appropriate security functional testing that demonstrates the TSF satisfies the security functional requirements.	O.THOROUGH_FUNCTIONAL_ TESTING (ATE_FUN.1, ATE_COV.2, ATE_DPT.2, ATE_IND.2) ensures that adequate functional testing is performed to ensure the TSF satisfies the security functional requirements and demonstrates that the TOE's security mechanisms operate as documented. While functional testing serves an important purpose, it does not ensure the TSFI cannot be used in unintended ways to circumvent the TOE's security policies.
	O.VULNERABILITY_ANALYSIS_ TEST The TOE will undergo appropriate independent vulnerability analysis and penetration testing to demonstrate the design and implementation of the TOE does not allow attackers with medium attack potential to violate the TOE's security policies.	O.VULNERABILITY_ANALYSIS_TEST (AVA_VAN.3, ADV_ARC.1, ADV_FSP.4, ADV_TDS.3) requires that the TOE be developed using sound engineering principles. By accurately and completely documenting the design of the security mechanisms in the TOE, including a security model, the design of the TOE can be better understood, which increases the chances that design errors will be discovered. The design of the TOE is independently analyzed for design flaws. Having an independent party perform the assessment ensures an objective approach is taken and may find errors in the design that would be left undiscovered by developers that have a preconceived incorrect understanding of the TOE's design. Since self-protection is a property of the TSF that is achieved through the design of the TOE and TSF, and enforced by the correct implementation of that design, self-protection will be achieved by that design and implementation.

T.RESIDUAL_DATA A user or process may gain unauthorized access to data through reallocation of TOE resources from one user or process to another.	O.RESIDUAL_INFORMATION The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated.	O.RESIDUAL_INFORMATION (FDP_RIP.1(1), FDP_RIP.1(2), FCS_CKM.4) counters this threat by ensuring that TSF data and user data is not persistent when resources are released by one user/process and allocated to another user/process. This means that network packets will not have residual data from another packet due to the padding of a packet.
T.RESOURCE_EXHAUSTION A malicious process or user may block others from TOE system resources (e.g., connection state tables) via a resource exhaustion denial of service attack.	O.RESOURCE_SHARING The TOE shall provide mechanisms that mitigate attempts to exhaust connection-oriented resources provided by the TOE (e.g., entries in a connection state table; TCP connections used by proxies).	O.RESOURCE_SHARING (FRU_RSA.1, FMT_MTD.2, FMT_MOF.1, FRU_PRS.1) mitigates this threat by requiring the TOE to provide controls over connection-oriented resources. These controls provide the administrator ability to specify which network identifiers have access to the TOE's connection-oriented resources over a time period that is specified by the administrator. This objective also addresses the denial-of-service attack of a user attempting to exhaust the connection-oriented resources by generating a large number of half-open connections (e.g., SYN attack).
T.SPOOFING An entity may mis-represent itself as the TOE to obtain authentication data.	O.TRUSTED_PATH The TOE will provide a means to ensure users are not communicating with some other entity pretending to be the TOE, and that the TOE is communicating with an authorized IT entity and not some other entity pretending to be an authorized IT entity.	O.TRUSTED_PATH (FTP_TRP.1, FTP_ITC.1) mitigates this threat by ensuring users have the capability to ensure they are communicating with the TOE when providing identification and authentication data to the TOE.

T.MALICIOUS_TSF_COMPROMISE A malicious user or process may cause TSF data or executable code to be inappropriately accessed (viewed, modified, or deleted).	O.DISPLAY_BANNER The TOE will display an advisory warning regarding use of the TOE.	O.DISPLAY_BANNER (FTA_TAB.1) helps mitigate this threat by providing the Security Administrator the ability to remove product information (e.g., product name, version number) from a banner that is displayed to users. Having product information about the TOE provides an attacker with information that may increase their ability to compromise the TOE.
	O.MANAGE The TOE will provide all the functions and facilities necessary to support the administrators in their management of the security of the TOE, and restrict these functions and facilities from unauthorized use.	O.MANAGE (FMT_MTD.1, FMT_MTD.2, FMT_MSA.1, FMT_MSA.3, FMT_MOF.1, FMT_SMF.1, FAU_SAR.1, FAU_SAR.2, FAU_SAR.3, FAU_STG.1, FAU_STG.3, FAU_STG.4, FAU_ARP_EXT.1) is necessary because an access control policy is not specified to control access to TSF data. This objective is used to dictate who is able to view and modify TSF data, as well as the behavior of TSF functions.
	O.RESIDUAL_INFORMATION The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated.	O.RESIDUAL_INFORMATION (FDP_RIP.1(1), FDP_RIP.1(2), FCS_CKM.4) is necessary to mitigate this threat, because even if the security mechanisms do not allow a user to explicitly view TSF data, if TSF data were to inappropriately reside in a resource that was made available to a user, that user would be able to inappropriately view the TSF data.

	O.SELF_PROTECTION The TSF will maintain a domain for its own execution that protects itself and its resources from external interference, tampering, or unauthorized disclosure.	O.SELF_PROTECTION (ADV_ARC.1, FTP_ITC.1, FTP_TRP.1, FPT_FLS.1, FRU_FLT.1(1), FRU_FLT.1(2), FRU_FLT.1(3, FRU_FLT.2) contributes to countering this threat by ensuring that the TSF can protect itself from users. ADV_ARC.1 provides the security architecture description of the security domains maintained by the TSF that are consistent with the SFRs. Since self-protection is a property of the TSF that is achieved through the design of the TOE and TSF, and enforced by the correct implementation of that design, self-protection will be achieved by that design and implementation.
	O.TRUSTED_PATH The TOE will provide a means to ensure users are not communicating with some other entity pretending to be the TOE, and that the TOE is communicating with an authorized IT entity and not some other entity pretending to be an authorized IT entity.	O.TRUSTED_PATH (FTP_TRP.1, FTP_ITC.1) plays a role in addressing this threat by ensuring that a trusted communication path exists between the TOE and authorized users (i.e., remote administrators, authorized IT entities). This ensures the transmitted data cannot be compromised or disclosed (e.g., encrypted) during the duration of the trusted path. The protection offered by this objective is limited to TSF data and security attributes (e.g., proxy user's user data is not protected, since their entire session communication (only the authentication portion of the session is protected) does not require encryption or any other form of protection).

T.UNATTENDED_SESSION A user may gain unauthorized access to an unattended session.	O.ROBUST_TOE_ACCESS The TOE will provide mechanisms that control a user's logical access to the TOE and to explicitly deny access to specific users when appropriate	O.ROBUST_TOE_ACCESS (FTA_TSE.1, FIA_UID.2, FIA_SOS.2, FTA_SSL.3, AVA_VAN.3, FIA_AFL.1, FIA_ATD.1, FIA_UAU.1, FIA_UAU.5) helps to mitigate this threat by including mechanisms that place controls on user's sessions. Local administrator's sessions are locked and remote sessions are dropped after a Security Administrator defined time period of inactivity. Locking the local administrator's session reduces the opportunity of someone gaining unauthorized access the session when the console is unattended. Dropping the connection of a remote session (after the specified time period) reduces the risk of someone accessing the remote machine where the session was established, thus gaining unauthorized access to the session.
T.UNAUTHORIZED_ACCESS A user may gain access to services (by sending data through or to the TOE) for which they are not authorized according to the TOE security policy.	O.MEDIATE The TOE must mediate the flow of information between sets of TOE network interfaces or between a network interface and the TOE itself in accordance with its security policy.	O.MEDIATE (FDP_IFF.1, FDP_IFC.1, FMT_REV.1, ADV_ARC.1) works to mitigate this threat by ensuring that all network packets that flow through the TOE are subject to the information flow policies.
T.UNIDENTIFIED_ACTIONS The administrator may fail to notice potential security violations, thus limiting the administrator's ability to identify and take action against a possible security breach.	O.AUDIT_REVIEW The TOE will provide the capability to selectively view audit information, and alert the administrator of identified potential security violations.	O.AUDIT_REVIEW (FAU_SAA.1, FAU_ARP.1, FAU_ARP_EXT.1, FAU_SAR.1, FAU_SAR.3, FMT_MOF.1) helps to mitigate this threat by providing the Security Administrator with a required minimum set of configurable audit events that could indicate a potential security violation.

	O.ROBUST_ADMIN_GUIDANCE The TOE will provide administrators with the necessary information for secure delivery and management.	O.ROBUST_ADMIN_GUIDANCE (ALC_DEL.1, AGD_PRE.1, AGD_OPE.1) provides administrative guidance for the secure start-up of the TOE as well as guidance to configure and administer the TOE securely. This guidance provides administrators with the information necessary to ensure that the TOE is started and initialized in a secure manner. The guidance also provides information about the corrective measure necessary when a failure occurs (i.e., how to bring the TOE back into a secure state).
T.UNKNOWN_STATE When the TOE is initially started or restarted after a failure, design flaws, or improper configurations may cause the security state of the TOE to be unknown.	O.SOUND_DESIGN The design of the TOE will be the result of sound design principles and techniques; the design of the TOE, as well as the design principles and techniques, are adequately and accurately documented.	O.SOUND_DESIGN (ADV_FSP.4, ADV_TDS.3,) works to mitigate this threat by requiring that the TOE developers provide accurate and complete design documentation of the security mechanisms in the TOE. By providing this documentation, the possible security states of the TOE at startup or restart after failure should be documented and understood, thereby reducing the possibility that the TOE's security state could be unknown to users of the TOE.
	O.ROBUST_ADMIN_GUIDANCE The TOE will provide administrators with the necessary information for secure delivery and management.	O.ROBUST_ADMIN_GUIDANCE (ALC_DEL.1, AGD_PRE.1, AGD_OPE.1) provides administrative guidance for the secure start-up of the TOE as well as guidance to configure and administer the TOE securely. This guidance provides administrators with the information necessary to ensure that the TOE is started and initialized in a secure manner. The guidance also provides information about the corrective measure necessary when a failure occurs (i.e., how to bring the TOE back into a secure state).

P.ACCESS_BANNER The TOE shall display an initial banner describing restrictions of use, legal agreements, or any other appropriate information to which users consent by accessing the system.	O.DISPLAY_BANNER The TOE will display an advisory warning regarding use of the TOE.	O.DISPLAY_BANNER (FTA_TAB.1) satisfies this policy by ensuring that the TOE displays a Security Administrator configurable banner that provides all users with a warning about the unauthorized use of the TOE.
P.ACCOUNTABILITY The authorized users of the TOE shall be held accountable for their actions within the TOE.	O.AUDIT_GENERATION The TOE will provide the capability to detect and create records of security-relevant events associated with users.	O.AUDIT_GENERATION (FAU_GEN.1, FAU_GEN.2, FIA_USB.1, FAU_STG.3, FAU_STG.4) addresses this policy by providing the Security Administrator with the capability of configuring the audit mechanism to record the actions of a specific user, or review the audit trail based on the identity of the user. Additionally, the administrator's ID is recorded when any security relevant change is made to the TOE (e.g. access rule modification, start-stop of the audit mechanism, establishment of a trusted channel, etc.).
	O.TIME_STAMPS The TOE shall provide reliable time stamps and the capability for the administrator to set the time used for these time stamps.	O.TIME_STAMPS (FPT_STM.1, FMT_MTD.1) plays a role in supporting this policy by requiring the TOE to provide a reliable time stamp (configured locally by the Security Administrator or via an external NTP server). The audit mechanism is required to include the current date and time in each audit record. All audit records that include the user ID, will also include the date and time that the event occurred.

	O.ROBUST_TOE_ACCESS The TOE will provide mechanisms that control a user's logical access to the TOE and to explicitly deny access to specific users when appropriate	O.ROBUST_TOE_ACCESS (FTA_TSE.1, FIA_UID.2, FIA_SOS.2, FTA_SSL.3, AVA_VAN.3, FIA_AFL.1, FIA_ATD.1, FIA_UAU.1, FIA_UAU.5) supports this policy by requiring the TOE to identify and authenticate all authorized users prior to allowing any TOE access or any TOE mediated access on behalf of those users. While the user ID of authorized users can be assured, since they are authenticated, this PP allows unauthenticated users to access the TOE and the identity is then a presumed network identifier (e.g., IP address).
P.ADMIN_ACCESS Administrators shall be able to administer the TOE remotely through protected communications channels.	O.ADMIN_ROLE The TOE will provide an administrator role to isolate administrative actions.	O.ADMIN_ROLE (FMT_SMR.2) supports this policy by requiring the TOE to provide mechanisms (e.g., local authentication, remote authentication, means to configure and manage the TOE both remotely and locally) that allow remote and local administration of the TOE. This is not to say that everything that can be done by a local administrator must also be provided to the remote administrator. In fact, it may be desirable to have some functionality restricted to the local administrator (e.g., setting the ruleset).□
	O.TRUSTED_PATH The TOE will provide a means to ensure users are not communicating with some other entity pretending to be the TOE, and that the TOE is communicating with an authorized IT entity and not some other entity pretending to be an authorized IT entity.	O.TRUSTED_PATH (FTP_TRP.1, FTP_ITC.1) satisfies this policy by requiring that each remote administrative session (all administrative roles) is authenticated and conducted via a secure channel. Additionally, all authorized IT entities (e.g. authentication/certificate servers, NTP servers) must adhere to the same requirements as the remote administrator.

P.CRYPTOGRAPHIC_FUNCTIONS The TOE shall provide cryptographic functions for its own use, including encryption/decryption, key generation and destruction and cryptographic hashing services.	O.CRYPTOGRAPHIC_FUNCTIONS The TOE provides cryptographic functions for its own use, including encryption/decryption, key generation and destruction and cryptographic hashing services.	O.CRYPTOGRAPHIC_FUNCTIONS (FCS_CKM.1, FCS_CKM.4, FCS_COP.1(1), FCS_COP.1(2)) implements this policy, requiring cryptographic mechanisms that are used to provide encryption/decryption services. Functions include key generation and destruction, encryption, decryption and cryptographic hashing services.
P.VULNERABILITY_ ANALYSIS_TEST The TOE must undergo appropriate independent vulnerability analysis and penetration testing to demonstrate that the TOE is resistant to an attacker possessing a medium attack potential.	O.VULNERABILITY_ANALYSIS_ TEST The TOE will undergo appropriate independent vulnerability analysis and penetration testing to demonstrate the design and implementation of the TOE does not allow attackers with medium attack potential to violate the TOE's security policies.	O.VULNERABILITY_ANALYSIS_TEST (AVA_VAN.3, ADV_ARC.1, ADV_FSP.4, ADV_TDS.3) requires that the TOE be developed using sound engineering principles. By accurately and completely documenting the design of the security mechanisms in the TOE, including a security model, the design of the TOE can be better understood, which increases the chances that design errors will be discovered. The design of the TOE is independently analyzed for design flaws. Having an independent party perform the assessment ensures an objective approach is taken and may find errors in the design that would be left undiscovered by developers that have a preconceived incorrect understanding of the TOE's design. Since self-protection is a property of the TSF that is achieved through the design of the TOE and TSF, and enforced by the correct implementation of that design, self-protection will be achieved by that design and implementation.

Table 10-2: Threat to Objective Mapping

10.2 EAL 4 Justification

The threats that were chosen are consistent with attacker of medium attack potential, therefore EAL4 was chosen for this ST.

10.3 Requirement Dependency Rationale

All Security Functional Requirement component dependencies have been met by the TOE as defined by the CEM.

10.4 Security Functional Requirements Rationale

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

Objective	Security Functional Component	Rationale
O.ROBUST_ADMIN_GUIDANCE The TOE will provide administrators with the necessary information for secure delivery and management.	AGD_OPE.1 Operational User Guidance	AGD_OPE.1 describes the proper use of the TOE from a user standpoint.
	AGD_PRE.1 Preparative Procedures	AGD_PRE.1 documents the procedures necessary and describes the steps required for the secure installation, generation, and start-up of the TOE.
	ALC_DEL.1 Delivery Procedures	ALC_DEL.1 describes product delivery and a description of all procedures used to ensure objectives are not compromised in the delivery process.
O.ADMIN_ROLE The TOE will provide an administrator role to isolate administrative actions.	FMT_SMR.2 Restrictions on security roles	FMT_SMR.2 states that there will be a Security Administrator, Cryptographic Administrator and Audit Administrator on the TOE. All default roles are distinct and there will be no overlap of operations.
O.AUDIT_GENERATION The TOE will provide the capability to detect and create records of security-relevant events associated with users.	FAU_GEN.1 Audit Data Generation	FAU_GEN.1 states that the TSF shall be able to generate an audit record of the start-up and shutdown of the audit functions and all auditable events listed in Table 6-2 Auditable Events.
	FAU_GEN.2 User Identity Association	FAU_GEN.2 ensures that the audit records associate a user identity with the auditable event.

	FAU_STG.3 Action in case of possible audit data loss	FAU_STG.3 requires that the administrators are alerted when the audit trail exceeds a capacity threshold established by the Security Administrator. This ensures that the Security Administrator has the opportunity to manage the audit trail before it becomes full and the avoiding the possible loss of audit data.
	FAU_STG.4 Prevention of audit data loss	FAU_STG.4 states that the TSF shall overwrite the oldest stored audit records and immediately alert the administrators by displaying a message at the remote management console when an administrative session exists for each of the defined administrative roles if the audit trail is full.
	FIA_USB.1 User-Subject Binding	FIA_USB.1 plays a role is satisfying this objective by requiring a binding of security attributes associated with users that are authenticated with the subjects that represent them in the TOE. This only applies to authorized users, since the identity of unauthenticated users cannot be confirmed. Therefore, the audit trail may not always have the proper identity of the subject that causes an audit record to be generated (e.g., presumed network address of an unauthenticated user may be a spoofed address).
O.AUDIT_PROTECTION The TOE will provide the capability to protect audit information.	FAU_STG.1 Protected Audit Trail Storage	FAU_STG.1 states that the TSF shall be able to protect the audit trail and prevent unauthorized modifications to the stored audit records in the audit trail.
	FAU_SAR.2 Restricted Audit Review	FAU_SAR.2 restricts the ability to read the audit trail to the administrators, thus preventing the disclosure of the audit data to any other user.

	FAU_STG.3 Action in case of possible audit data loss	FAU_STG.3 requires that the administrators are alerted when the audit trail exceeds a capacity threshold established by the Security Administrator. This ensures that the Security Administrator has the opportunity to manage the audit trail before it becomes full and the avoiding the possible loss of audit data.
	FAU_STG.4 Prevention of audit data loss	FAU_STG.4 states that the TSF shall overwrite the oldest stored audit records and immediately alert the administrators by displaying a message at the remote management console when an administrative session exists for each of the defined administrative roles if the audit trail is full. Additionally, if so configured by the administrators, the TOE will enter Maintenance Mode if the log records are full and overwriting is disabled.
	FMT_MOF.1 Management of security functions behavior	FMT_MOF.1 restricts the capability to modify the behavior of the audit and alarm functions to the Security Administrator.
O.AUDIT_REVIEW The TOE will provide the capability to selectively view audit information, and alert the administrator of identified potential security violations.	FAU_SAA.1 Potential violation analysis	FAU_SAA.1 states that the TSF shall be able to apply and enforce the accumulation or combination of a Security Administrator specified number of authentication failures and a Security Administrator specified threshold for the audit trail known to indicate a potential security violation, the failure to automatically update the Category Code Database, when the audit trail is full and will overwrite, any failure of the TSF self-tests.
	FAU_ARP.1 Security alarms	FAU_ARP.1 states that the TSF immediately display an alarm message that identifies the potential security violation. Additionally, the audit record content associated with the event that generated the alarm shall be accessible.

	FAU_ARP_EXT.1 Security alarm acknowledgement	FAU_ARP_EXT.1 requires that the TSF shall immediately display an acknowledgement message at all remote administrator sessions that received the alarm, identifying: a reference to the potential security violation, a notice that it has been acknowledged, the time of the acknowledgement, and the user identifier that acknowledged the alarm upon the acknowledgement of a potential security violation by an administrator.
	FAU_SAR.1 Audit review	FAU_SAR.1 provides the administrators with the capability to read all the audit data contained in the audit trail. This requirement also mandates the audit information be presented in a manner that is suitable for the administrators to interpret the audit trail, which is subject to interpretation. It is expected that the audit information be presented in such a way that the administrators can examine an audit record and have the appropriate information presented together to facilitate the analysis of the audit review.
	FAU_SAR.3 Selectable Audit review	FAU_SAR.3 complements FAU_SAR.1 by providing the administrators the flexibility to specify criteria that can be used to search or sort the audit records residing in the audit trail.
	FMT_MOF.1 Management of security functions behavior	FMT_MOF.1.1 The TSF shall restrict the ability to perform the management functions as listed in Table 6-3 Management Functions of the TOE.

O.CHANGE_MANAGEMENT The configuration of, and all changes to, the TOE and its development evidence will be analyzed, tracked, and controlled throughout the TOE's development.	ALC_CMC.4 Authorization Controls	ALC_CMC.4 contributes to this objective by requiring the developer have a configuration management plan that describes how changes to the TOE and its evaluation deliverables are managed. The developer is also required to employ a configuration management system that operates in accordance with the CM plan and provides the capability to control who on the development staff can make changes to the TOE and its developed evidence. This requirement also ensures that authorized changes to the TOE have been analyzed and the developer's acceptance plan describes how this analysis is performed and how decisions to incorporate the changes to the TOE are made. ALC_CMC.4 ALSO requires that the CM system use an automated means to control changes made to the TOE. If automated tools are used by the developer to analyze, or track changes made to the TOE, those automated tools must be described. This aids in understanding how the CM system enforces the control over changes made to the TOE.
	ALC_CMS.4 CM Scope	ALC_CMS.4 is necessary to define what items must be under the control of the CM system. This requirement ensures that the TOE implementation representation, design documentation, test documentation (including the executable test suite), user and administrator guidance, CM documentation and security flaws are tracked by the CM system.

	ALC_DVS.1 Identification of Security Measures	ALC_DVS.1 requires the developer describe the security measures they employ to ensure the integrity and confidentiality of the TOE are maintained. The physical, procedural, and personnel security measures the developer uses provides an added level of control over who and how changes are made to the TOE and its associated evidence.
	ALC_FLR.2 Flaw Reporting Procedures	ALC_FLR.2 plays a role in satisfying the "analyzed" portion of this objective by requiring the developer to have procedures that address flaws that have been discovered in the product, either through developer actions (e.g., developer testing) or those discovered by others. The flaw remediation process used by the developer corrects any discovered flaws and performs an analysis to ensure new flaws are not created while fixing the discovered flaws.
	ALC_LCD.1 Life-cycle Definition	ALC_LCD.1 requires the developer to document the life-cycle model used in the development and maintenance of the TOE. This life-cycle model describes the procedural aspects regarding the development of the TOE, such as design methods, code or documentation reviews, how changes to the TOE are reviewed and accepted or rejected.
O.CORRECT_TSF_OPERATION The TOE will provide the capability to test the TSF to ensure the correct operation of the TSF in its operational environment.	FPT_TST.1 TSF Testing	FPT_TST.1.1 states that the TSF shall run a suite of self tests during initial start-up, periodically during normal operation, at the request of the authorized user to demonstrate the correct operation of the TSF.
O.CRYPTOGRAPHIC_FUNCTIONS The TOE shall provide cryptographic functions for its own use, including	FCS_CKM.1 Cryptographic Key Generation	FCS_CKM.1 states that the TSF shall generate cryptographic keys using RSA with 2048 bit keys.

encryption/decryption, key generation and destruction and cryptographic hashing services Functions include key generation and destruction, encryption, decryption and cryptographic hashing services.	FCS_CKM.4 Cryptographic Key destruction	FCS_CKM.4 states that the TSF shall destroy keys with the overwrite method using no standard.
	FCS_COP.1(1) Cryptographic Operation	FCS_COP.1(1) states that the TSF shall perform encryption and decryption using AES with 256 bit keys.
	FCS_COP.1(2) Cryptographic Operation	FCS_COP.1.1(2) states that the TSF shall perform cryptographic hashing services using SHA-1 with 160 bit keys and SHA-256 with 256 bit keys..
O.DISPLAY_BANNER The TOE will display an advisory warning regarding use of the TOE.	FTA_TAB.1 Default TOE Access Banners	FTA_TAB.1 meets this objective by requiring the TOE display a Security Administrator defined banner before a user can establish an authenticated session. This banner is under complete control of the Security Administrator in which they specify any warnings regarding unauthorized use of the TOE and remove any product or version information if they desire.
O.THOROUGH_FUNCTIONAL_ TESTING The TOE will undergo appropriate security functional testing that demonstrates the TSF satisfies the security functional requirements.	ATE_COV.2 Analysis of coverage	ATE_COV.2 requires the developer to provide a test coverage analysis that demonstrates the TSFI are completely addressed by the developer's test suite.
	ATE_FUN.1 Functional Tests	ATE_FUN.1 requires the developer to provide the necessary test documentation to allow for an independent analysis of the developer's security functional test coverage.
	ATE_DPT.2 Testing: Security Enforcing modules	ATE_DPT.2 ensures that subtleties in TSF behavior that are not readily apparent in the functional specification are addressed in the test suite.
	ATE_IND.2 Independent Testing	ATE_IND.2 requires an independent confirmation of the developer's test results, by mandating a subset of the test suite be run by an independent party.

O.MANAGE The TOE will provide all the functions and facilities necessary to support the administrators in their management of the security of the TOE, and restrict these functions and facilities from unauthorized use.	FMT_MSA.1 Management of security attributes	The FMT requirements are used to satisfy this management objective, as well as other objectives that specify the control of functionality. The requirement's rationale for this objective focuses on the administrator's capability to perform management functions in order to control the behavior of security functions.
	FMT_MSA.3 Static attribute initialization	FMT_MSA.3 states that the TSF shall enforce the unauthenticated information flow SFP to provide restrictive default values for security attributes that are used to enforce the SFP. The TSF shall allow the Security Administrator to specify alternative initial values to override the default values when an object or information is created.
	FMT_MOF.1 Management of security functions behavior	FMT_MOF.1.1 The TSF shall restrict the ability to perform the management functions as listed in Table 6-3 Management Functions of the TOE.
	FMT_MTD.1 Management of TSF data	FMT_MTD.1 States that the TSF shall perform the management functions as listed in Table 6-4 Management of TSF Data.
	FMT_MTD.2 Management of limits on TSF data	FMT_MTD.2 states that the TSF shall restrict the specification of the limits for quotas on transport-layer connections and controlled connection-oriented resources to the Security Administrator. Once the quotas are met or exceeded, the TSF shall drop all packets above the quota.
	FMT_SMF.1 Specification of management functions	FMT_SMF.1 states that the TSF shall be capable of performing the management functions as listed in Table 6-4 Management of TSF Data.

	FAU_SAR.1 Audit review	FAU_SAR.1 ensures that the Audit Administrator has the capability to review the audit records and that they are presented in a manner that is suitable for review (e.g., the Audit Administrator can construct a sequence of events provided the necessary events were audited).
	FAU_SAR.2 Restricted audit review	FAU_SAR.2 restricts the ability to read the audit records to the administrators. This capability exists for the Security and Crypto administrators to help facilitate any trouble shooting that they may have to perform.
	FAU_SAR.3 Selectable audit review	FAU_SAR.3 provides the administrators with the ability to selectively review the contents of the audit trail based on established criteria. This capability allows the administrators to focus their audit review to what is pertinent at that time.
	FAU_STG.1 Protected audit trail storage	FAU_STG.1 specifies that only the Audit Administrator can delete the audit trail. This prevents the accidental or intentional deletion of the audit trail by administrators acting in another role.
	FAU_STG.3 Action in case of possible audit data loss	FAU_STG.3 provides the Security Administrator the capability to establish a threshold of audit trail capacity, that when reached an alarm will be generated.
	FAU_STG.4 Prevention of audit data loss	If the audit trail becomes full FAU_STG.4 provides the Security Administrator the option of having the TOE prevent auditable events from occurring, or having the TOE overwrite the oldest audit records. While the option of overwriting old audit records does not technically prevent audit data loss, it is provided to the Security Administrator as an option to prevent a possible denial-of-service.

	FAU_ARP_EXT.1 Security alarm acknowledgement	FAU_ARP_EXT.1 contributes to this objective in that it requires the administrators to acknowledge an alarm before it is no longer displayed. Without this requirement an alarm display message may be overwritten or lost without an administrator being aware of the alarm condition.
O.MEDIATE The TOE must mediate the flow of information between sets of TOE network interfaces or between a network interface and the TOE itself in accordance with its security policy.	FDP_IFF.1 Simple security attributes	FDP_IFF.1 states that the TSF shall enforce the unauthenticated information flow SFP based on the following types of subject and information security attributes :Source subject security attributes: set of source subject identifiers, Destination subject security attributes: Set of destination subject identifiers, Information security attributes: presumed identity of source subject; identity of destination subject; transport layer protocol; services; destination subject service identifier (e.g., TCP or UDP destination port number); category code for external network traffic; Stateful packet attributes.
	FDP_IFC.1 Subset information flow control	FDP_IFC.1 defines the subjects, information (e.g., objects) and the operations that are performed with respect to the three information flow policies.
	FMT_REV.1 Revocation	FMT_REV.1 is a management requirement that affords the Security Administrator the ability to immediately revoke user’s ability to send network traffic through the TOE. If the Security Administrator revokes a user’s access (e.g., via a rule in the ruleset, revoking an administrative role from a user) the TOE will immediately enforce the new Security Administrator defined “policy”.

	ADV_ARC.1 Security architecture description	ADV_ARC.1 contributes to countering this threat by ensuring that the TSF can protect itself from users. ADV_ARC.1 provides the security architecture description of the security domains maintained by the TSF that are consistent with the SFRs. Since self-protection is a property of the TSF that is achieved through the design of the TOE and TSF, and enforced by the correct implementation of that design, self-protection will be achieved by that design and implementation.
O.RESIDUAL_INFORMATION The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated.	FCS_CKM.4 Cryptographic key destruction	FCS_CKM.4 applies to the destruction of cryptographic keys used by the TSF. This requirement specifies how and when cryptographic keys must be destroyed. The proper destruction of these keys is critical in ensuring the content of these keys cannot possibly be disclosed when a resource is reallocated to a user.
	FDP_RIP.1 (1) Subset residual information protection	FDP_RIP.1(1) states that the TSF shall ensure that any previous information content of a resource is made unavailable upon the allocation of the resource to/deallocation of the resource from the kernel level objects.
	FDP_RIP.1 (2) Subset residual information protection	FDP_RIP.1.1 (2) states that the TSF shall ensure that any previous information content of a resource is made unavailable upon the deallocation of the resource from the user-space program level.
O.RESOURCE SHARING The TOE shall provide mechanisms that mitigate attempts to exhaust connection-oriented resources provided by the TOE (e.g., entries in a connection state table; Transmission Control Protocol	FRU_RSA.1 Maximum quotas	FRU_RSA.1 states that the TSF shall enforce maximum quotas on transport layer representation, controlled connection-oriented resources that subjects can use simultaneously.

(TCP) connections used by proxies).	FMT_MTD.2 Management of limits on TSF data	FMT_MTD.2 states that the TSF shall restrict the specification of the limits for quotas on transport-layer connections and controlled connection-oriented resources to the Security Administrator. Once the quotas are met or exceeded, the TSF shall drop all packets above the quota.
	FMT_MOF.1 Management of security functions behavior	FMT_MOF.1 The TSF shall restrict the ability to perform the management functions as listed in Table 6-3 Management Functions of the TOE.
	FPT_PRS.1 Limited priority of service	FPT_PRS.1 states that the TSF shall assign a priority to each subject and that the unauthenticated flow control is mediated on the basis of subject's assigned priority.
O.SELF_PROTECTION The TSF will maintain a domain for its own execution that protects itself and its resources from external interference, tampering, or unauthorized disclosure.	ADV_ARC.1 Security architecture description	ADV_ARC.1 contributes to countering this threat by ensuring that the TSF can protect itself from users. ADV_ARC.1 provides the security architecture description of the security domains maintained by the TSF that are consistent with the SFRs. Since self-protection is a property of the TSF that is achieved through the design of the TOE and TSF, and enforced by the correct implementation of that design, self-protection will be achieved by that design and implementation.
	FTP_ITC.1 Inter-TSF trusted channel	FTP_ITC.1.1 states that the TSF shall provide a communication channel between itself and another trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure. The trusted channel is used for updating system time, SNMP, and the Category Code Database].

	FTP_TRP.1 Trusted path	FTP_TRP.1.1 states that the TSF shall provide a communication path between itself and remote users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from modification and disclosure. The trusted path is used for initial user authentication and all administrative actions.
	FPT_FLS.1 Failure with preservation of secure state	FPT_FLS.1.1 states that the TSF shall preserve a secure state when any number of the following modules go down: Auto Update, PoliWall Process, msglogd, syslogd, pktlogd, pktlog6d.
	FRU_FLT.1(1) Degraded fault tolerance	FRU_FLT.1.1 (1) states that the TSF shall ensure the operation of information flow control when the Auto Update module goes down.
	FRU_FLT.1(2) Degraded fault tolerance	FRU_FLT.1.1 (2) states that the TSF shall ensure the operation of remote administration functions and access control when the PoliWall Process module goes down.
	FRU_FLT.1(3) Degraded fault tolerance	FRU_FLT.1.1 (3) states that the TSF shall ensure the operation of auditing functions when any number of the following auditing modules go down: msglogd, syslogd, pktlogd, pktlog6d.
	FRU_FLT.2 Limited fault tolerance	FRU_FLT.2.1 states that the TSF shall ensure the operation of all the TOE's capabilities when any number of the following modules go down: Auto Update, PoliWall Process, msglogd, syslogd, pktlogd, pktlog6d.

O.TIME_STAMPS The TOE shall provide reliable time stamps and the capability for the administrator to set the time used for these time stamps.	FPT_STM.1 Reliable time stamps	FPT_STM.1 requires that the TOE be able to provide reliable time stamps for its own use and therefore, partially satisfies this objective. Time stamps include date and time and are reliable in that they are always available to the TOE, and the clock must be monotonically increasing.
	FMT_MTD.1 Management of TSF data	FMT_MTD.1 States that the TSF shall perform the management functions as listed in Table 6-4 Management of TSF Data.
O.ROBUST_TOE_ACCESS The TOE will provide mechanisms that control a user's logical access to the TOE and to explicitly deny access to specific users when appropriate.	FTA_TSE.1 TOE session establishment	FTA_TSE.1.1 contributes to this objective by limiting a user's ability to logically access the TOE. This requirement provides the Security Administrator the ability to control when (e.g., time and day(s) of the week) and where (e.g., from a specific network address) remote administrators, as well as authorized IT entities can access the TOE.
	FIA_UID.2 User identification before any action	FIA_UID.2 plays a small role in satisfying this objective by ensuring that every user is identified before the TOE performs any mediated functions. In some cases, the identification cannot be authenticated (e.g., a user attempting to send a data packet through the TOE that does not require authentication; in which case the identity is presumed to be authentic). In other cases (e.g., administrators, and authorized IT entities), the identity of the user is authenticated. It is impractical to require authentication of all users that attempt to send data through the TOE, therefore, the requirements specified in the TOE require authentication where it is deemed necessary. This does impose some risk that a data packet was sent from an identity other than specified in the data packet.

	FIA_SOS.2 TSF Generation of secrets	FIA_SOS.2 states that the TSF shall be able to enforce the use of TSF generated secrets for authentication and access control.
	FTA_SSL.3 TSF-initiated termination	FTA_SSL.3 takes into account remote sessions. After a Security Administrator defined time interval of inactivity remote sessions will be terminated, this refers to remote administrative sessions. This component is especially necessary, since remote sessions are not typically afforded the same physical protections that local sessions are provided.
	AVA_VAN.3 Vulnerability analysis	AVA_VAN.3 The evaluator performs penetration testing, to confirm that the potential vulnerabilities cannot be exploited in the operational environment for the TOE. Penetration testing is performed by the evaluator assuming an attack potential of moderate. This requirement ensures the evaluator has performed an analysis of the authentication mechanism to ensure the probability of guessing a user's authentication data would require a high-attack potential,

	FIA_AFL.1 Authentication failure handling	FIA_AFL.1 provides a detection mechanism for unsuccessful authentication attempts by remote administrators and authorized IT entities. The requirement enables a Security Administrator settable threshold that prevents unauthorized users from gaining access to authorized user's account by guessing authentication data by locking the targeted account until the Security Administrator takes some action (e.g., re-enables the account) or for some Security Administrator defined time period. Thus, limiting an unauthorized user's ability to gain unauthorized access to the TOE.
	FIA_ATD.1 User attribute definition	FIA_ATD.1 defines the attributes of users, including a userid that is used to by the TOE to determine a user's identity and enforce what type of access the user has to the TOE (e.g., the TOE associates a userid with any role(s) they may assume). This requirement allows a human user to have more than one user identity assigned, so that a single human user could assume all the roles necessary to manage the TOE. In order to ensure a separation of roles, this PP requires a single role to be associated with a user id. This is inconvenient in that the administrator would be required to log in with a different user id each time they wish to assume a different role, but this helps mitigate the risk that could occur if an administrator were to execute malicious code.

	FIA_UAU.1 Timing of authentication	FIA_UAU.1 contributes to this objective by limiting the services that are provided by the TOE to unauthenticated users. Management requirements and the unauthenticated information flow policy requirement provide additional control on these services.
	FIA_UAU.5 Multiple authentication mechanisms	FIA_UAU.5 states that the TSF shall provide username/password or username/password with client certificate and the TSF shall authenticate any user's claimed identity according to the Security Administrators configurable.
O.TRUSTED_PATH The TOE will provide a means to ensure administrators are not communicating with some other entity pretending to be the TOE, and that the TOE is communicating with an authorized IT entity and not some other entity pretending to be an authorized IT entity.	FTP_ITC.1 Inter-TSF trusted channel	FTP_ITC.1 is similar to FTP_TRP.1 in that it requires a mechanism that creates a distinct communication path with the same characteristics, however FTP_ITC.1 is used to protect communications between IT entities, rather than between a human user and an IT entity. FTP_ITC.1.3 requires the TOE to initiate the trusted channel, which ensures that the TOE has established a communication path with an authorized IT entity and not some other entity pretending to be an authorized IT entity.

	FTP_TRP.1 Trusted path	FTP_TRP.1.1 requires the TOE to provide a mechanism that creates a distinct communication path that protects the data that traverses this path from disclosure or modification. This requirement ensures that the TOE can identify the end points and ensures that a user cannot insert themselves between the user and the TOE, by requiring that the means used for invoking the communication path cannot be intercepted and allow a “man-in-the-middle-attack” (this does not prevent someone from capturing the traffic and replaying it at a later time – see FPT_RPL.1). Since the user invokes the trusted path (FTP_TRP.1.2) mechanism they can be assured they are communicating with the TOE. FTP_TRP.1.3 mandates that the trusted path be the only means available for providing identification and authentication information, therefore ensuring a user’s authentication data will not be compromised when performing authentication functions. Furthermore, the remote administrator’s communication path is encrypted during the entire session.

O.VULNERABILITY_ANALYSIS TEST The design of the TOE will be the result of sound design principles and techniques; the design of the TOE, as well as the design principles and techniques, are adequately and accurately documented.	AVA_VAN.3 Vulnerability analysis	The AVA_VAN.3 component provides the necessary level of confidence that vulnerabilities do not exist in the TOE that could cause the security policies to be violated. AVA_VAN.3 requires the evaluator to perform a search of public domain sources to identify potential vulnerabilities in the TOE. The evaluator will perform an independent, methodical vulnerability analysis of the TOE using the guidance documentation, functional specification, TOE design, security architecture description and implementation representation to identify potential vulnerabilities in the TOE. The evaluator will conduct penetration testing based on the identified potential vulnerabilities to determine that the TOE is resistant to attacks performed by an attacker possessing enhanced-basic attack potential.
	ADV_FSP.4 Functional Specification with complete summary	The functional specification will completely represent the TSF, describe the purpose and method of use for all TSFI, identify and describe all parameters associated with each TSFI, describe all actions associated with each TSFI and describe all direct error messages that may result from an invocation of each TSFI. The tracing will demonstrate that the functional requirements trace to TSFIs in the functional specification. Any processing that is externally visible performed by NIC must be specified in the functional specification. Having a complete understanding of what is available at the TSF interface allows one to analyze this functionality in the context of design flaws.

	ADV_TDS.3 Architectural Design	The design will describe: the structure of the TOE in terms of subsystems; the TSF in terms of modules; identify all subsystems of the TSF; provide a description of each subsystem of the TSF; a description of the interactions among all subsystems of the TSF; a mapping from the subsystems of the TSF to the modules of the TSF; describe each SFR-enforcing module in terms of its purpose; describe each SFR-enforcing module in terms of its SFR-related interfaces; return values from those interfaces, and called interfaces to other modules; describe each SFR-supporting or SFR-non-interfering module in terms of its purpose and interaction with other modules; the mapping shall demonstrate that all behavior described in the TOE design is mapped to the TSFIs that invoke it.
	ADV_ARC.1 Security Architecture Description	ADV_ARC.1 ensures that the TSF can protect itself from users and provides the security architecture description of the security domains maintained by the TSF that are consistent with the SFRs. Since self-protection is a property of the TSF that is achieved through the design of the TOE and TSF, and enforced by the correct implementation of that design, self-protection will be achieved by that design and implementation.
OE.CRYPTANALYTIC Cryptographic methods used in the IT environment shall be interoperable with the TOE, and should be resistant to cryptanalytic attacks (i.e., will be of adequate strength to protect unclassified Mission Support, Administrative, or Mission Critical data).	FTP_ITC.1 Inter-TSF trusted channel	FPT_ITC.1 ensures that encryption is used on the communication channel between authorized IT entities and the TOE.

	FTP_TRP.1 Trusted path	FPT_TRP.1 ensures that an administrator can be assured that they are communicating with the TOE.

Table 10-3: Security Functional Requirements Rationale

11 Assurance Measures

This section identifies the assurance measures provided by the developer in order to meet the security assurance requirement components for EAL4 augmented with ASE_TSS.2 and ALC_FLR.2. A description of each of the TOE assurance measures follows in Table 1-23.

Component	Document(s)	Rationale
ADV_ARC.1 Security Architecture Design	• TOE Design Specification for TechGuard Security PoliWall version 0.6 • LLD.zip	This document describes the security architecture of the TOE.
ADV_FSP.4 Functional Specification with complete summary	Functional Specification Document for TechGuard Security PoliWall version 0.7	This document describes the functional specification of the TOE with complete summary.
ADV_IMP.1 Implementation Representation of the TSF	Source Code Files.zip	This document describes the implementation of the TOE.
ADV_TDS.3 Architectural Design	• TOE Design Specification for TechGuard Security PoliWall version 0.6 • LLD.zip	This document describes the architectural design of the TOE.
AGD_OPE.1 Operational User Guidance	• PoliWall CCF Users Manual.pdf • Poliwall-CCF Quick Start Guide v2-01-01.pdf	This document describes the operational user guidance for.
AGD_PRE.1 Preparative Procedures	• PoliWall CCF Users Manual.pdf • Poliwall-CCF Quick Start Guide v2-01-01.pdf	This document describes the preparative procedures that need to be done prior to installing.
ALC_CMC.4 Authorizations Controls	PoliWall Configuration Management Capabilities Documentation v 0.5	This document describes the authorization controls for the TOE.
ALC_CMS.4 CM Scope	• cctl_software_item_list.txt • cctl_software_item_list_no_kernels.txt • FogBugzSecurityFlawScreenshot.jpg • subversion_tag_list_output.txt • PoliWall Configuration Management Scope	These documents describe the CM scope of the TOE.

Component	Document(s)	Rationale
	Documentation v0.4 • cctl_software_item_list_old version.txt • Logs.zip	
ALC_DEL.1 Delivery Procedures	PoliWall Delivery Documentation v 0.3.	This document describes product delivery for and a description of all procedures used to ensure objectives are not compromised in the delivery process.
ALC_DVS.1 Identification of Security Measures	• Training Plan version ORG-0001-003 • Process & Product Quality Assurance version SUP-0002-003 • TECHGUARD SECURITY® SECURITY POLICY revision May 30, 2007	This document provides an identification of security measures for the TOE.
ALC_FLR.2 Flaw reporting procedures	PoliWall Flaw Remediation Document v 0.2	This document provides the policies for issuing new releases of the TOE as corrective actions.
ALC_LCD.1 Life-Cycle Definition	Project Planning and Management version PM-0001-005	This document provides the life-cycle definition of the TOE.
ALC_TAT.1 Tools and Techniques	PoliWall Tools and Techniques Documentation version 0.4	This document describes the tools and techniques used in the life cycle development of the TOE.
ASE_CCL.1 Conformance Claims	TechGuard Security PoliWall Security Target version 0.6	This document describes the CC conformance claims made by the TOE.
ASE_ECD.1 Extended Components Definition	TechGuard Security PoliWall Security Target version 0.6	This document provides a definition for all extended components in the TOE.
ASE_INT.1 Security Target Introduction	TechGuard Security PoliWall Security Target version 0.6	This document describes the Introduction of the Security Target.
ASE_OBJ.2 Security Objectives	TechGuard Security PoliWall Security Target version 0.6	This document describes all of the security objectives for the TOE.
ASE_REQ.2 Security Requirements	TechGuard Security PoliWall Security Target version 0.6	This document describes all of the security requirements for the TOE.
ASE_SPD.1 Security Problem Definition	TechGuard Security PoliWall Security Target version 0.6	This document describes the security problem definition of the Security Target.
ASE_TSS.2 TOE Summary Specification	TechGuard Security PoliWall Security Target version 0.6	This document describes the TSS section of the Security

Component	Document(s)	Rationale
		Target.
ATE_COV.2 Analysis of Coverage	• TechGuard Final Testing 20110125.zip • Testing Overview.doc	This document provides an analysis of coverage for the TOE.
ATE_DPT.2 Testing: Security enforcing modules	• TechGuard Final Testing 20110125.zip • Testing Overview.doc	This document describes the security enforcing modules of the TOE.
ATE_FUN.1 Functional Tests	• TechGuard Final Testing 20110125.zip • Testing Overview.doc	This document describes the functional tests for the TOE.
ATE_IND.2 Independent Testing	• TechGuard Security PoliWall Evaluation Team Test Report version 1.0 • Booz Allen_TechGuard_PoliWall_INDTestProcedures.xlsx	This document describes the independent testing for the TOE.
AVA_VAN.3 Vulnerability Analysis	TechGuard Security PoliWall v2.01.01 version 1.0	This document describes the vulnerability analysis of the TOE.

Table 11-1: Assurance Requirements Evidence