Aggregation Services Router (ASR) 1000 Series

Security Target

Revision 0.19

May 2011

Table of Contents

List of Tables

DOCUMENT INTRODUCTION

Prepared By:
Cisco Systems, Inc.
170 West Tasman Dr.
San Jose, CA 95134

This document provides the basis for an evaluation of a specific Target of Evaluation (TOE), the Aggregation Services Router (ASR) 1000 Series. This Security Target (ST) defines a set of assumptions about the aspects of the environment, a list of threats that the product intends to counter, a set of security objectives, a set of security requirements, and the IT security functions provided by the TOE which meet the set of requirements.

REVISION HISTORY

Rev	Date	Description
0.01	February 2009	Initial Draft
0.02	March 2009	Updated for BU comments
0.03	May 2009	Updated for Lab comments
0.04	May 2009	Updated for Lab comments 2
0.05	June 2009	Updated for BU comments
0.06	June 2009	Updated for BU clarifications
0.07	July 2009	General editing: formatting, typos. Expanded acronym table.
0.08	July 2009	Updated to address iVOR comments
0.09	July 2009	Updated to include SPAs in TOE boundary
0.10	Oct. 2009	Updated to address VOR comments
0.11	June 2010	Updated to address ADV ETR verdicts
0.12	June 2010	Updated for consistency with testing and OPE
0.13	September 2010	Updated for AGD submission
0.14	October 2010	Updated for testing, ADV ETR verdicts
0.15	October 2010	Updated for AGD ETR
0.16	January 2011	Updated for ATE findings
0.17	March 2011	Updated for ATE findings
0.18	April 2011	Updated for the ATE ETR
0.19	May 2011	Updated for tVOR

1 SECURITY TARGET INTRODUCTION

The Security Target contains the following sections:

- Security Target Introduction [Section 1]
- Conformance Claims [Section 2]
- Security Problem Definition [Section 3]
- Security Objectives [Section 4]
- IT Security Requirements [Section 5]
- TOE Summary Specification [Section 6]
- Rationale [Section 7]

The structure and content of this ST comply with the requirements specified in the Common Criteria (CC), Part 1, Annex A, and Part 3, Chapter 4.

1.1 ST and TOE Reference

This section provides information needed to identify and control this ST and its TOE. This ST targets Medium Robustness.

Table 1: ST and TOE Identification

ST Title	Cisco Aggregation Services Router (ASR) 1000 Series Security Target
ST Version	0.19
Publication Date	May 2011
Vendor and ST Author	Cisco Systems, Inc.
TOE Reference	Cisco Aggregation Services Router (ASR) 1000 Series
TOE Hardware Models	ASR 1002f, ASR 1002, ASR 1004, ASR 1006
TOE Software Version	IOS XE 2.4.2t
ST Evaluation Status	In Evaluation
Keywords	Switch, Data Protection, Authentication, VPN, Router, Firewall, Encryption

1.2 Acronyms and Abbreviations

The following acronyms and abbreviations are used in this Security Target:

Table 2: Acronyms

Acronyms / Abbreviations	Definition
AAA	Administration, Authorization, and Accounting
ACK	Acknowledgement
ACL	Access Control List
AES	Advanced Encryption Standard
ANSI	American National Standards Institute
ASR	Aggregation Services Router
BGP	Border Gateway Protocol
CC	Common Criteria for Information Technology Security Evaluation
CCEVS	Common Criteria Evaluation and Validation Scheme
CCM	Counter with Cipher Block Chaining-Message Authentication Code
CCMB	Common Criteria Maintenance Board
CEM	Common Evaluation Methodology for Information Technology Security

Acronyms / Abbreviations	Definition
CLI	Command Line Interface
CM	Configuration Management
CRL	Certificate and Certificate Revocation List
CSP	Critical Security Parameter
DH	Diffie-Hellman
DHCP	Dynamic Host Configuration Protocol
EAL	Evaluation Assurance Level
EEPROM	Electrically Erasable Programmable Read-Only Memory
ESP	Embedded Services Processor
FIPS	Federal Information Processing Standard
GDOI	Group Domain of Interpretation
HMAC	Hashed Message Authentication Code
HTTPS	Hyper-Text Transport Protocol Secure
ICMP	Internet Control Message Protocol
IOS	Internet Operating System
ISSU	In-Service Software Upgrade
IKE	Internet Key Exchange
IP	Internet Protocol
IPSec	IP Security
IS-IS	Intermediate System-to-Intermediate System
IT	Information Technology
KDF	Key Derivation Function
MAC	Message Authentication Code
MPLS	Multiprotocol Label Switching
MRPP	Medium Robustness Protection Profile
NIC	Network Interface Card
NIST	National Institute of Standards and Technology
OS	Operating System
OCSP	Online Certificate Status Protocol
OSP	Organizational Security Policy
OSPF	Open Shortest Path First
PIM	Protocol Independent Multicast
POS	Packet Over Sonet
PP	Protection Profile
pp_fw_tf_mr_v1.1	U.S. Government Protection Profile for Traffic Filter Firewall For Medium Robustness Environments
pp_router_mr_v1.1	U.S. Government Router Protection Profile For Medium Robustness Environments
pp_vpn_mr_v1.2	U.S. Government Virtual Private Network (VPN) Boundary Gateway Protection Profile For Medium Robustness Environments
PRNG	Pseudo Random Number Generator
QoS	Quality of Service
rDSA	RSA Digital Signature Algorithm
RFC	Request for Comment
RIP	Routing Information Protocol
RNG	Random Number Generator
RP	Route Processor
RSA	Rivest, Shamir, and Adleman
RU	Rack Unit
SA	Security Association
SEQ	Sequence
SFP	Security Function Policy
SFR	Security Functional Requirement

Acronyms / Abbreviations	Definition
SHS	Secure Hash Standard
SPA	Shared Port Adaptor
SSH	Secure Shell
ST	Security Target
TCP	Transport Control Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
TDEA	Triple Data Encryption Algorithm
TDES	Triple Data Encryption Standard
TOE	Target of Evaluation
TSC	TSF Scope of Control
TSF	TOE Security Function
TLS	Transport Layer Security
TSFI	TOE Security Function Interface
TSP	TOE Security Policy
UDP	User Datagram Protocol
VLAN	Virtual Local Area Network
VPN	Virtual Private Network
VRF	Virtual Routing and Forwarding
WAN	Wide Area Network

1.3 TOE Overview

The TOE is a purpose-built, wide-area network (WAN) routing platform that includes firewall and VPN functionality. The TOE includes four (4) chassis options, ASR 1002, ASR 1002f, ASR 1004, and ASR 1006.

1.3.1 TOE Product Type

The Cisco ASR 1000 Series Router (ASR 1002, ASR 1002f, ASR 1004, ASR 1006) delivers embedded hardware acceleration for multiple Cisco IOS® XE Software services. In addition, the Cisco ASR 1000 Series Router features redundant Route and Embedded Services Processors, as well as software-based redundancy.

In support of the routing capabilities, the Cisco ASR 1000 Series Router provides IPSec connection capabilities for VPN enabled clients connecting through the Cisco ASR 1000 Series Router. The Cisco ASR 1000 Series Router also supports firewall capabilities. The ASR 1000 Series Router is a single-device security and routing solution for protecting the WAN entry point into the network. Zone-based firewall allows grouping of physical and virtual interfaces into zones to simplify logical network topology. The creation of these zones facilitates the application of firewall policies on a zone-to-zone basis, instead of having to configure policies separately on each interface.

1.3.2 Supported non-TOE Hardware/ Software/ Firmware

The TOE supports (in some cases optionally) the following hardware, software, and firmware in its environment:

Table 3: IT Environment Components

IT Environment Component	Required	Usage/Purpose Description for TOE performance
VPN Peer	No	This includes any peer with which the TOE participates in VPN communications. VPN peers may be any device or software client that supports IPSec communications. Both VPN clients and VPN gateways are considered VPN peers by the TOE.

1.4 TOE DESCRIPTION

This section provides an overview of the Aggregation Services Router (ASR) 1000 Series Target of Evaluation (TOE). This section also defines the TOE components included in the evaluated configuration of the TOE. The TOE consists of a number of components including:

- Chassis: The TOE chassis includes 2-RU, 4-RU, and 6-RU form factors. The chassis is the component of the TOE in which all other TOE components are housed.
- Embedded Services Processor (ESP): The Cisco ASR 1000 Series ESPs are responsible for the data-plane processing tasks, and all network traffic flows through them.
- Route Processor (RP): The Cisco ASR 1000 Series RPs provide the advanced routing capabilities of the TOE. They also monitor and manage the other components in the Cisco ASR 1000 Series Aggregation Services.
- Shared Port Adaptors (SPAs): Used for connecting to networks. These SPAs interface with the TOE to provide the network interfaces that will be used to communicate on the network.

Table 4: TOE Component Descriptions

Hardware Model	Cisco ASR 1002f	Cisco ASR 1002	Cisco ASR 1004	Cisco ASR 1006
Software Model	Cisco IOS XE Version 2.4.2t	Cisco IOS XE Version 2.4.2t	Cisco IOS XE Version 2.4.2t	Cisco IOS XE Version 2.4.2t
				
Size	2-Rack Units	2-Rack Units	4-Rack Units	6-Rack Units
Power	DC power: 590W AC Power: 560W	DC power: 590W AC Power: 560W	DC power: 1020W AC Power: 960W	DC power: 1700W AC Power: 1600W
Supported ESPs	Integrated ESP	ESP5 ESP10	ESP10 ESP20	Dual ESP10 Dual ESP20
Supported ESP Throughput	2.5 Gbps	ESP5 – 5 Gbps ESP10 – 10 Gbps	ESP10 – 10 Gbps ESP20 – 20 Gbps	ESP10 – 10 Gbps ESP20 – 20 Gbps
Supported ESP Processors	Freescale 8543	Freescale 8543 (both ESPs)	Freescale 8543 (both ESPs)	Freescale 8543 (both ESPs)
Supported RPs	Integrated RP	Integrated RP	RP1 RP2	Dual RP1 Dual RP2

Supported RP Processors	Freescale 8548	Freescale 8548	RP1: Freescale 8548 RP2: Intel Wolfdale-DP	RP1: Freescale 8548 RP2: Intel Wolfdale-DP
Supported SPAs (all TOE model support all SPAs)	Cisco 8-Port Channelized T1/E1 Shared Port Adapter (SPA-8XCHT1/E1) Cisco 4-Port Channelized T3 (DS0) Shared Port Adapter (SPA-4XCT3/DS0) Cisco 2-Port Channelized T3 (DS0) Shared Port Adapter (SPA-2XCT3/DS0) Cisco 1-port Channelized STM-1/OC-3c to DS0 Shared Port Adapter (SPA-1XCHSTM1/OC3) Cisco 2-Port Clear Channel T3/E3 Shared Port Adapter (SPA-2XT3/E3) Cisco 4-Port Clear Channel T3/E3 Shared Port Adapter (SPA-4XT3/E3) Cisco 4-Port Serial Interface Shared Port Adapter (SPA-4XT-Serial) Cisco 4-Port 10BASE-T/100BASE Fast Ethernet Shared Port Adapter (SPA-4X1FE-TX-V2) Cisco 8-Port 10BASE-T/100BASE Fast Ethernet Shared Port Adapter (SPA-8X1FE-TX-V2) Cisco 2-Port Gigabit Ethernet Shared Port Adapter (SPA-2X1GE-V2) Cisco 5-Port Gigabit Ethernet Shared Port Adapter (SPA-5X1GE-V2) Cisco 8-Port Gigabit Ethernet Shared Port Adapter (SPA-8X1GE-V2) Cisco 10-Port Gigabit Ethernet Shared Port Adapter (SPA-10X1GE-V2) Cisco 1-Port 10 Gigabit Ethernet Shared Port Adapter (SPA-1X10GE-L-V2) Cisco 2-Port OC3c/STM-1c POS Shared Port Adapter (SPA-2XOC3-POS) Cisco 4-Port OC3c/STM-1c POS Shared Port Adapter (SPA-4XOC3-POS) Cisco 8-port OC3/STM4 POS Shared Port Adapter (SPA-8XOC3-POS) Cisco 1-Port OC12c/STM-4c POS Shared Port Adapter (SPA-1XOC12-POS) Cisco 2-port OC12/STM4 POS Shared Port Adapter (SPA-2XOC12-POS) Cisco 4-port OC12/STM4 POS Shared Port Adapter (SPA-4XOC12-POS) Cisco 8-port OC12/STM4 POS SPA Shared Port Adapter (SPA-8XOC12-POS) Cisco 1-port OC48/STM16 POS/RPR Shared Port Adapter (SPA-1XOC48-POS/RPR) Cisco 2-port OC48/STM16 POS/RPR Shared Port Adapter (SPA-2XOC48POS/RPR) Cisco 4-port OC48/STM16 POS/RPR Shared Port Adapter (SPA-4XOC48POS/RPR) Cisco 1-Port OC-192c/STM-64c POS/RPR Shared Port Adapter with XFP Optics (SPA-OC192POS-XFP)			
SPA Slots	1 SPA slot	3 SPA slots	8 SPA slots	12 SPA slots
Interfaces	Port Adapter Interface Console Port Auxiliary Port 10/100 BITS Ethernet Port 10/100 Management Ethernet Port USB Port GigE Ports (4)	Port Adapter Interface (3) Console Port Auxiliary Port 10/100 BITS Ethernet Port 10/100 Management Ethernet Port USB Port GigE Ports (4)	Port Adapter Interface (8) Console Port Auxiliary Port 10/100 Management Ethernet Port 10/100 BITS Ethernet Port (1) USB Ports (2)	Port Adapter Interface (12) Console Port Auxiliary Port (2) 10/100 BITS Ethernet Port (2) 10/100 Management Ethernet Port (2) USB Ports (4)
Hardware Redundancy Supported?	Not supported	Not supported	Not supported	Supported

1.5 Physical Scope of the TOE

The TOE is a hardware and software solution that makes up the Aggregation Services Router (ASR) 1000 Series Router. The TOE is comprised of the following:

Table 5: Physical Scope of the TOE

TOE Configuration	Hardware Configurations	Software Version
ASR 1002f	No configuration options	IOS XE 2.4.2t software running on a hardened version of Linux Kernel 2.6.8.
ASR 1002	ESP5 or ESP10	IOS XE 2.4.2t software running on a hardened version of Linux Kernel 2.6.8.
ASR 1004	RP 1 or RP 2 ESP10 or ESP20	IOS XE 2.4.2t software running on a hardened version of Linux Kernel 2.6.8.
ASR 1006	RP 1 or RP 2 Dual ESP10 or ESP20	IOS XE 2.4.2t software running on a hardened version of Linux Kernel 2.6.8.

As identified above, there are several configurations available for each TOE hardware model (ASR 1002, ASR 1002f, ASR 1004, ASR 1006). Each model supports one or more Embedded Services Processors (ESP) and one or more Router Processors (RP). Additionally, each TOE hardware model is configured to include one or more SPAs to facilitate network connectivity. The following table identifies the number of SPAs supported by each TOE hardware model and identifies the SPAs included within the TOE.

Table 6: Physical Scope of the TOE

TOE Configuration	SPA Slots	TOE SPAs
ASR 1002f	1 SPA slot	Cisco 8-Port Channelized T1/E1 Shared Port Adapter Cisco 4-Port Channelized T3 (DS0) Shared Port Adapter Cisco 2-Port Channelized T3 (DS0) Shared Port Adapter Cisco 1-port Channelized STM-1/OC-3c to DS0 Shared Port Adapter Cisco 2-Port Clear Channel T3/E3 Shared Port Adapter Cisco 4-Port Clear Channel T3/E3 Shared Port Adapter Cisco 4-Port Serial Interface Shared Port Adapter Cisco 4-Port 10BASE-T/100BASE Fast Ethernet Shared Port Adapter Cisco 8-Port 10BASE-T/100BASE Fast Ethernet Shared Port Adapter Cisco 2-Port Gigabit Ethernet Shared Port Adapter Cisco 5-Port Gigabit Ethernet Shared Port Adapter Cisco 8-Port Gigabit Ethernet Shared Port Adapter Cisco 10-Port Gigabit Ethernet Shared Port Adapter Cisco 1-Port 10 Gigabit Ethernet Shared Port Adapter Cisco 2-Port OC3c/STM-1c POS Shared Port Adapter Cisco 4-Port OC3c/STM-1c POS Shared Port Adapter Cisco 8-port OC3/STM4 POS Shared Port Adapter
ASR 1002	3 SPA slot	
ASR 1004	8 SPA slot	
ASR 1006	12 SPA slot	

		Cisco 1-Port OC12c/STM-4c POS Shared Port Adapter Cisco 2-port OC12/STM4 POS Shared Port Adapter Cisco 4-port OC12/STM4 POS Shared Port Adapter Cisco 8-port OC12/STM4 POS SPA Shared Port Adapter Cisco 1-port OC48/STM16 POS/RPR Shared Port Adapter Cisco 2-port OC48/STM16 POS/RPR Shared Port Adapter Cisco 4-port OC48/STM16 POS/RPR Shared Port Adapter Cisco 1-Port OC-192c/STM-64c POS/RPR Shared Port Adapter with XFP Optics

The following provides a functional description of each sub-component.

1.5.1 Embedded Services Processor (5Gbps, 10Gbps, 20Gbps)

The ESPs are responsible for the data-plane processing tasks, and all network traffic flows through them. Packets arrive to the ESPs from the network. Each packet is decoded, interpreted, processed and forwarded, as necessary, by the ESP. The ESP performs all baseline packet routing operations, including MAC classification, Layer 2 and Layer 3 forwarding, quality-of-service (QoS) classification, policing and shaping, security access control lists (ACLs), VPN, load balancing, and NetFlow. The ESPs contain a cryptographic co-processor. This co-processor is dedicated to providing cryptographic acceleration for cryptographic operations within the ASR 1000.

1.5.2 Route Processor (RP1, RP2)

The RPs within the ASR 1000 provides the advanced packet routing capabilities of the ASR 1000 Series Router. The RPs provide the monitoring, managing, and configuring services for the TOE itself. All TOE administration is performed within the RPs. The administrative CLI interface is provided by the Route Processors. The RPs also negotiate and maintain IPSec authentication, encryption methods, and encryption keys between the TOE and external IT entities.
As noted in table 4, there are embedded RPs in the lower models and RP1 and RP2's in the higher models. The RP1 are first-generation Cisco ASR 1000 Series Route Processors and the RP2's are second-generation Cisco ASR 1000 Series Route Processors. These RPs interact with a separate Freescale processor for offloading some cryptographic computations.

1.5.3 Shared Port Adaptors

SPAs provide the physical interfaces for TOE connectivity to the connected network including copper, channelized, POS, and Ethernet.

1.6 Logical Scope of the TOE

The TOE is comprised of several security features. Each of the security features identified above consists of several security functionalities, as identified below.

1. Identification & Authentication

2. Security Management
3. VPN, Router, and/or Firewall Information Flow Control
4. Trusted Channel/Path
5. Cryptography
6. Security Audit
7. Availability

These features are described in more detail in the subsections below.

1.6.1 Identification & Authentication

The ASR performs two types of authentication: device-level authentication of the remote device (VPN peers) and user authentication for the Authorized Administrator of the ASR. Device-level authentication allows the ASR to establish a secure channel with a trusted peer. The secure channel is established only after each device authenticates itself. Device-level authentication is performed via IKE/IPSec mutual authentication. The ASR provides authentication services for administrative users wishing to connect to the ASRs secure CLI administrative interface. The TOE requires authorized administrators to authenticate prior to being granted access to any of the management functionality.

1.6.2 Security Management

The TOE provides secure administrative services for management of general TOE configuration and the security functionality provided by the TOE. All TOE administration occurs either through a secure SSHv2 session via terminal server or via a local console connection. The TOE provides the ability to securely manage all TOE administrative users; all audit functionality of the TOE; all TOE cryptographic functionality; and the information flow control policies enforced by the TOE. The TOE supports three separate administrative roles: Cryptographic Administrator, Audit Administrator and Security Administrator. The Cryptographic Administrator is responsible for the configuration and maintenance of cryptographic elements related to the establishment of secure connections to and from the TOE. The Audit Administrator is responsible for the regular review of the TOE's audit data. The Security Administrator is responsible for all other administrative tasks.

When an administrative session is initially established, the TOE displays a Security Administrator configurable warning banner. This is used to provide any information deemed necessary by the Security Administrator. The TOE supports several scenarios in which the administrative session is either locked out or terminated, as follows;

- The TOE allows an administrator to lock out her administrative session on demand.
- The TOE locks administrative sessions based on a configured period of inactivity.
- The TOE terminates the administrative session after a configurable time interval of session inactivity occurs.

1.6.3 VPN, Router, and/or Firewall Information Flow Control

The TOE enforces several information flow control policies, including:

- VPN services
- Unauthenticated TOE services
- Unauthenticated information flow

Each of these enforced information flows are further discussed below.

1.6.3.1 VPN services

The VPN process includes remote device authentication, negotiation of specific cryptographic parameters for the session, and providing a secure connection from and to the remote device. For inbound or outbound connections with external IT entities that are capable of supporting VPN (e.g., a peer ASR 1000 series router, a VPN Peer), the TOE will establish a secure connection. For other inbound or outbound traffic a secure connection will not be established.

1.6.3.2 Unauthenticated TOE services

The Cisco ASR 1000 Series Routers mediate all information flows to and from the ASR itself. The TOE has the ability to permit or deny information flows based on the characteristics of the information flow. By examining the information flows to the TOE itself, the ASR is able to provide specific TOE services to requesting unauthenticated entities. The TOE services that are available to unauthenticated entities are configurable by the Security Administrator and must include, ICMP. All other TOE services are only available to authenticated entities.

1.6.3.3 Unauthenticated information flow

The Cisco ASR 1000 mediates all information flows through the ASR for unauthenticated information flows. The TOE provides the ability to classify all data flows into zones. Configurable allow or deny rule sets are applied to each information flow on a zone by zone basis. All security attributes are inspected based on the configurable rule set of the information flow. The TOE makes the decision to allow or deny unauthenticated information flows based on the configured information flow rule set. The ASR generates and maintains “state” information for all approved connections mediated by the TOE. The “state” information is used to monitor the status of an approved connection and validate incoming packets received as part of an approved connection.

The TOE ensures that all information flows from the TOE do not contain residual information from previous traffic. Packets are padded with a specific defined pattern. Residual data is never transmitted from the TOE. Additionally, The TOE maintains counters of the number of the connections through the TOE. When the TOE’s counters

exceed the maximum sessions, the TOE will take actions to reduce the number of connections.

1.6.4 Trusted Path/Channel

The TOE establishes a trusted path between the TOE and the remote management station used by the administrators to manage the TOE. This Trusted path is secured using an SSHv2 secure connection. All remote administration occurs through the SSHv2 secure trusted path. Alternatively, the TOE supports local administration through a directly connected management station.

The ASR establishes a trusted channel between itself and peer IT devices. Between the ASR and peer routers, network control information is exchanged via trusted channels to allow dynamic connection establishment and packet routing. Network control information consists of specific requests and instructions that include destination address, routing controls, and signaling information. Trusted channels are secured via IPSec encryption.

1.6.5 Cryptography

The TOE provides cryptography in support of other ASR security functionality. This cryptography has been validated for conformance to the requirements of FIPS 140-2 overall Level 2 and Level 3 for sections 3 and 10. The ASR provides cryptography in support of VPN connections. The cryptographic services provided by the TOE in support of IPSec include:

Table 7: IPSec Related Cryptography

Cryptographic Method	Use within IPSec
Internet Key Exchange	Used to establish initial IPSec session.
SP 800-56 Key Exchange	Used in IPSec session establishment.
Group Domain of Interpretation	Used in IPSec session establishment.
RSA Digital Signatures	Used in IPSec session establishment.
ANSI X9.31 PRNG	Used in IPSec session establishment.
SHS	Used to provide IPSec traffic integrity verification.
AES	Used to encrypt IPSec session traffic.

The TOE also provides cryptography in support of secure administration. The following table identifies the cryptography provided in support of the secure administration.

Table 8: SSHv2 Related Cryptography

Cryptographic Method	Use within SSHv2
SP 800-56 Key Exchange	Used in SSHv2 session establishment.
RSA Digital Signatures	Used in SSHv2 session establishment.
ANSI X9.31 PRNG	Used in SSHv2 session establishment.
SHS	Used to provide SSHv2 traffic integrity verification.
AES	Used to encrypt SSHv2 session traffic.

NOTE: See the entries for FCS_CKM.1(1), FCS_CKM.1(2), FCS_CKM_(EXT).2, FCS_CKM.4, FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3), FCS_COP.1(4), and FCS_COP_(EXT).1 within section 6.1, "TOE Security Functional Requirement Measures" for additional details regarding the use of cryptography within SSHv2.

In support of the provided cryptography, the TOE performs a number of self-tests to ensure the correct operation. These tests include,

- Self tests to demonstrate the correct operation of the following cryptographic functions:
  - Key error detection;
  - cryptographic algorithms;
  - RNG/PRNG
- Self tests to demonstrate the correct operation of each key generation component
- Self tests to verify the integrity of TSF data related to the key generation
- Self tests to verify the integrity of stored TSF executable code
- Self tests to demonstrate the correct operation of the TSF

1.6.6 Security Audit

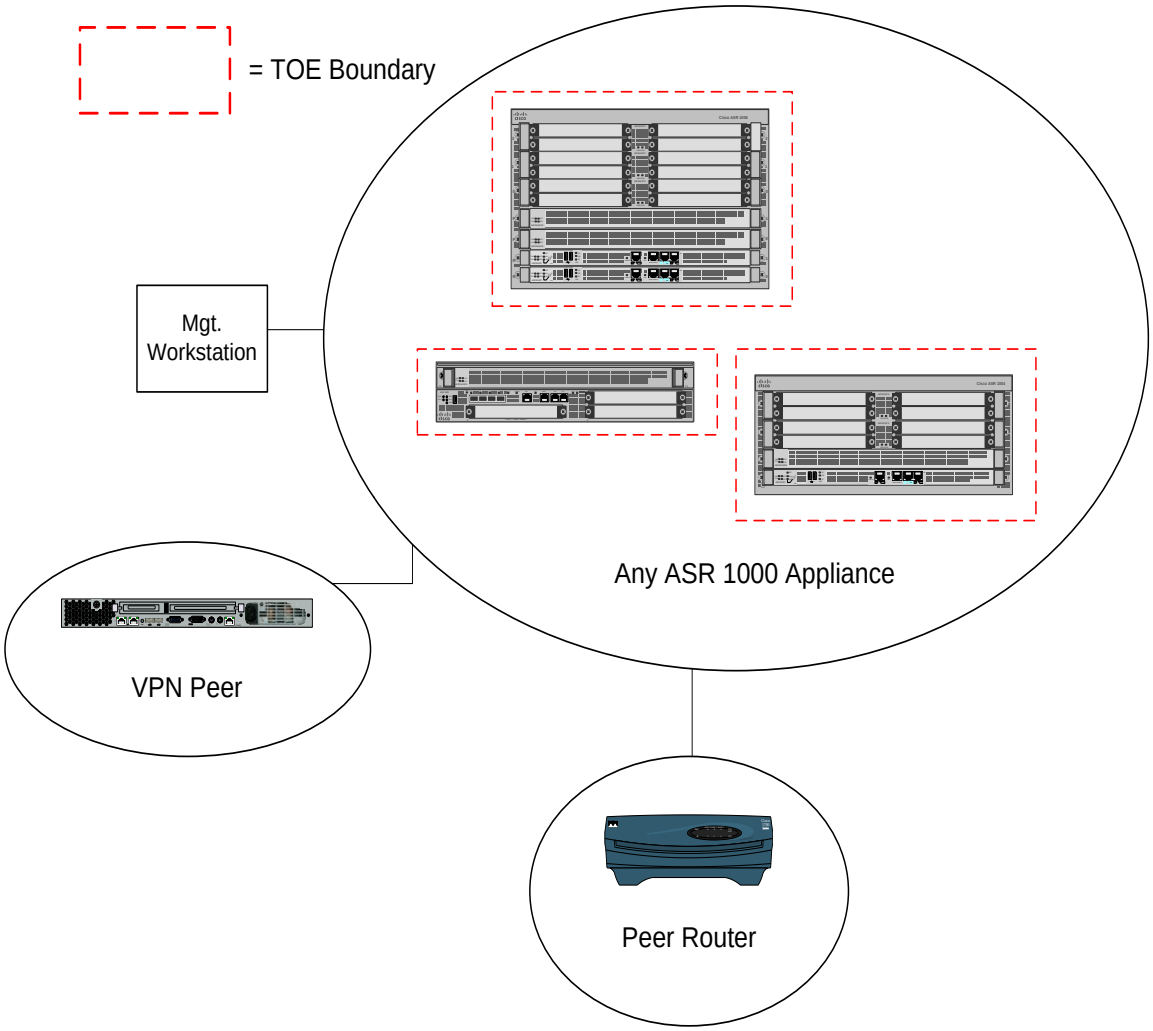
The ASR provides extensive auditing capabilities. The TOE can audit events related to security alarms, cryptographic functionality, information flow control enforcement, identification and authentication, and administrative actions. The ASR generates an audit record for each auditable event. In addition to generating audit records for auditable events, the TOE monitors the occurrences and identifies potential security violation based on the generated audit records. Once the ASR has detected a potential security violation, an alarm is generated and a message is displayed to administrators. Additionally, the Security Administrator can configure the TOE to generate an audible alarm to indicate a potential security violation and enforces confirmation of each alarm by an administrator. The ASR provides the Audit Administrator with a sorting and searching capability to improve audit analysis. The Security Administrator configures auditable events, backs-up and manages audit data storage. The TOE provides the Security Administrator with a circular audit trail or a configurable audit trail threshold to track the storage capacity of the audit trail.

1.6.7 High Availability

For ASR configurations that include dual ESPs or RPs, one of the ESPs or RPs act as the active hardware while the other acts as a hot standby. If there is a hardware failure within either the active ESP or active RP, the hot standby ESP or RP within the ASR automatically becomes active. If there is a software failure within the active software instance, the ASR automatically switches to the hot standby software instance resident within the TOE on the hot standby.

1.7 TOE Evaluated Configuration

The following figure provides a visual depiction of an example TOE deployment. The TOE boundary is surrounded with a hashed red line.



The previous figure includes the following:

- Several examples of TOE Models
  - ASR 1002
  - ASR 1004
  - ASR 1006
- VPN Peer (IT Environment)
- Peer Router (IT Environment)
- Management Workstation

NOTE: While the previous figure includes three TOE devices and three non-TOE IT environment devices, the TOE is only the ASR 1000 device. Only one TOE device is required for deployment of the TOE in an evaluated configuration.

1.7.1 Excluded Functionality

The following functional is excluded from the evaluation.

Table 9: Excluded Functionality

Excluded Functionality	Exclusion Rationale
Dual IOS mode – dual instances on a single ASR 1000 of the system software	This functionality provides software redundancy within the TOE. Software redundancy is not a security functionality required by the Protection Profiles for which conformance is claimed.
In-Service Software Upgrade (ISSU)	This functionality provides the ability to upgrade the TOE software without taking the TOE out of commission. The functionality is not a security functionality required by the Protection Profiles for which conformance is claimed.
Any TLS communication with the TOE	TLS communications with the TOE were excluded from FIPS 140-2 validations. These types of connections would include HTTPS connections with external servers. The TOE does not require any communication with external servers via HTTPS to provide the functionality described in the ST.
SNMP and Web User Interface management	These management interfaces do not enforce the required role privileges.
Access to the Linux shell within the ASR 1000 Series router	The Linux shell access could be used to execute other (non-TOE) applications within the router. Note that access to this shell has been removed from the TOE software image.
The physical auxilary port, the BITS Ethernet Port, and the USB port.	They have no current use with the TOE.
External NTP server	The TOE must rely upon its own internal timestamp per the PP requirements.
External Authentication server	The TOE must rely upon local authentication mechanisms per the PP requirements.
Management via telnet and ftp	These protocols send authentication data in the clear.
Usage of debug.conf	The FIPS 140-2 validation restricts usage of the debug.conf file to set environment variable values.
Level-based privilege separation in IOS XE.	This evaluation creates custom non-hierarchical roles that are not level-based but command access based.

These services may be disabled by configuration (with the exception of the Linux shell, and physical ports which are disabled in the TOE software and cannot be enabled). The exclusion of this functionality does not affect compliance to any of the identified Protection Profiles.

2 CONFORMANCE CLAIMS

2.1 Common Criteria Conformance Claim

The TOE and ST are compliant with the Common Criteria (CC) Version 3.1, Revision 3, dated: July 2009.

The TOE and ST are EAL4 Augmented with ALC_FLR.2 Part 3 conformant.
The TOE and ST are CC Part 2 extended.

2.2 Protection Profile Conformance

This ST claims functional compliance to the following Common Criteria validated Protection Profiles:

Table 10: Protection Profiles

Protection Profile	Version	Date
U.S. Government Router Protection Profile For Medium Robustness Environments	1.1	July 25, 2007
U.S. Government Virtual Private Network (VPN) Boundary Gateway Protection Profile For Medium Robustness Environments	1.2	Jan 30, 2009
U.S. Government Protection Profile for Traffic Filter Firewall For Medium Robustness Environments	1.1	July 25, 2007

2.2.1 Protection Profile Refinements

The three Protection Profiles for which conformance is claimed contain several SFRs, Threats, Assumptions, OSPs, and Objectives which convey the same functionality but are worded slightly differently. For each SFR, Threat, Assumption, OSP, and Objective, the text was combined within the Security Target.

2.2.2 Protection Profile Additions

The following threats were added to the TOE:

- T.NORECOVERY

The following objectives were added to the TOE:

- O.HA

The following requirements were added to the set of SFRs on the TOE:

- FCS_GDOI_(EXT).1,
- FPT_HA_(EXT).1

2.3 Protection Profile Conformance Claim Rationale

2.3.1 TOE Appropriateness

The ASR TOE provides all of the Routing, VPN, and Traffic Filter Firewall functionality at a level of security comensurate with that identified in the U.S. Government Protection Profiles:

- U.S. Government Router Protection Profile For Medium Robustness Environments (pp_router_mr_v1.1)
- U.S. Government Virtual Private Network (VPN) Boundary Gateway Protection Profile For Medium Robustness Environments (pp_vpn_mr_v1.2)
- U.S. Government Protection Profile for Traffic Filter Firewall For Medium Robustness Environments (pp_fw_tf_mr_v1.1)
  Note that strict compliance with these protection profiles is not being claimed, but in accordance with guidance from CCEVS, demonstrable conformance is being claimed for the assurance requirements in the PPs.

2.3.2 TOE Security Problem Definition Consistency

The Assumptions, Threats, and Organization Security Policies included in the Security Target represent a combination of the Assumptions, Threats, and Organization Security Policies specified in the three Protection Profiles for which conformance is claimed. All concepts covered in each of the Protection Profile's Security Problem Definitions are included in the Security Target. The following table identifies each assumption included in the ST and provides rationale for its inclusion in the Security Target with regards to the claims Protection Profiles.

Table 11: Assumption Protection Profile Conformance

Assumption	PPs	Rationale
A.NO_GENERAL_PURPOSE	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	The PPs contain two variants of the Assumption that provide the same information. The version found in pp_router_mr_v1.1 and pp_fw_tf_mr_v1.1 was used because they are slightly more comprehensive.
A.PHYSICAL	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	The PPs contain two variants of the Assumption that provide the same information. The version found in pp_vpn_mr_v1.2 and pp_fw_tf_mr_v1.1 was used for readability.
A.NO_TOE_BYPASS	pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Replicated exactly from PPs
A.AVAILABILITY	pp_router_mr_v1.1	Replicated exactly from PP

The following table identifies each OSP included in the ST and provides rationale for its inclusion in the Security Target with regards to the claims Protection Profiles.

Table 12: OSP Protection Profile Conformance

OSP	PPs	Rationale
P.CRYPTOGRAPHIC_FUNCTIONS	pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
P.CRYPTOGRAPHY_VALIDATED	pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
P.VULNERABILITY_ANALYSIS_TEST	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
P.INTEGRITY	pp_vpn_mr_v1.2	Reproduced exactly from the PP
P.ACCESS_BANNER	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	The PPs contained two versions of the OSP that were different by only one word (substituting "TOE" for "system"). The version found in pp_router_mr_v1.1 was used for readability.
P.ACCOUNTABILITY	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
P.ADMIN_ACCESS	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
P.CRYPTOGRAPHY	pp_router_mr_v1.1	Reproduced exactly from the PP
P.COMPATIBILITY	pp_router_mr_v1.1	Reproduced exactly from the PP

The following table identifies each Threat included in the ST and provides rationale for its inclusion in the Security Target with regards to the claims Protection Profiles.

Table 13: Threat Protection Profile Conformance

Threat	PPs	Rationale
T.ADMIN_ERROR	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
T.UNAUTHORIZED_ACCESS	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Used the variant of the Threat found in pp_fw_tf_mr_v1.1 as the base text because it is the most comprehensive version of the Threat. Combined the reference to User Data found in pp_router_mr_v1.1 for completeness. No threat was not addressed in the reproduction.
T.ADDRESS_MASQUERADE	pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
T.AUDIT_COMPROMISE	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
T.CRYPTO_COMPROMISE	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	The PPs contained two slightly different variants of this threat. The variant found in pp_router_mr_v1.1 was used for readability.
T.FLAWED_DESIGN	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
T.FLAWED_IMPLEMENTATION	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs

Threat	PPs	Rationale
T.MALICIOUS_TSF_ COMPROMISE	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
T.POOR_TEST	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced the variation of the Threat found in pp_router_mr_v1.1 and pp_vpn_mr_v1.2 because it is the most comprehensive version of the Threat.
T.REPLAY	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	The PPs contain two very similar versions of the Threat. The version found in pp_vpn_mr_v1.2 and pp_fw_tf_mr_v1.1 for readability.
T.RESIDUAL_DATA	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
T.UNAUTHORIZED_PEER	pp_router_mr_v1.1 pp_vpn_mr_v1.2	Reproduced exactly from the PPs
T.UNIDENTIFIED_ACTIONS	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
T.UNKNOWN_STATE	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	There are several variations of this threat found in the PPs. The variant found in pp_fw_tf_mr_v1.1 was used because the functionality is described in the most detail in that version of the threat. There is no contradictory information in this variation of the threat.
T.ADMIN_ROGUE	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced the variation of the threat found in pp_router_mr_v1.1. The text is the same as the other variations with the exception that the acronym is spelled out.
T.MASQUERADE	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced the variation of the threat found in pp_router_mr_v1.1 because the text covers everything found in the other variations of the threat plus additional information.
T.RESOURCE_EXHAUSTION	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced the variation of the threat found in pp_router_mr_v1.1 because the text identifies an additional type of attach to the type of attacks found in the other variations of the threat.
T.SPOOFING	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced the variation of the threat found in pp_router_mr_v1.1 because the text identifying the type of attackers is more specific then the text in the other variations of the Threat. Nothing is excluded from the variation that was reproduced.
T.UNATTENDED_SESSION	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
T.EAVESDROP	pp_router_mr_v1.1	Reproduced exactly from the PP
T.NORECOVERY	New Threat	This threat addresses high availability found in the TOE. This does not contradict any of the functionality found in the other threats.

2.3.3 Statement of Security Objectives Consistency

The Security Objectives included in the Security Target represent a combination of the Security Objectives specified in the three Protection Profiles for which conformance is claimed. All concepts covered in each of the Protection Profile's Statement of Security Objectives are included in the Security Target. The following table identifies each objective included in the ST and provides rationale for its inclusion in the Security Target with regards to the claims Protection Profiles.

Table 14: Objective Protection Profile Conformance

Objective	PPs	Rationale
O.ROBUST_ADMIN_GUIDANCE	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
O.AUDIT_GENERATION	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
O.AUDIT_PROTECTION	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
O.AUDIT_REVIEW	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
O.CHANGE_MANAGEMENT	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
O.CORRECT_ TSF_OPERATION	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
O.CRYPTOGRAPHIC_ FUNCTIONS	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Since the variant of this objective found in pp_router_mr_v1.1 contains all the information in the other variants of the objectives plus additional information, the variant found in pp_router_mr_v1.1 was reproduced in the ST
O.CRYPTOGRAPHY_ VALIDATED	pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
O.DISPLAY_BANNER	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
O.DOCUMENT_KEY_LEAKAGE	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Omitted as the required assurance requirements were not included.
O.SELF_PROTECTION	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
O.SOUND_DESIGN	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	This objective was identified in all three PPs. The variant in pp_router_mr_v1.1 contains the same information with slightly different wording. The variant included in pp_vpn_mr_v1.2 and pp_fw_tf_mr_v1.1 was used for readability.
O.SOUND_IMPLEMENTATION	pp_router_mr_v1.1	Reproduced exactly from the PPs

Objective	PPs	Rationale
	pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	
O.TIME_STAMPS	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
O.ROBUST_TOE_ACCESS	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
O.TRUSTED_PATH	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	The variant of the objective found in pp_vpn_mr_v1.2 was reproduced in the ST because it included the broadest language.
O.USER_GUIDANCE	pp_router_mr_v1.1	Reproduced exactly from the PP
O.VULNERABILITY_ANALYSIS_TEST	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
O.ADMIN_ROLE	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced the variant found in pp_router_mr_v1.1 and pp_vpn_mr_v1.2 because this variant contains all of the information in the other variant plus additional information.
O.INTEGRITY	pp_vpn_mr_v1.2	Reproduced exactly from the PP
O.MAINT_MODE	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
O.MANAGE	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
O.PEER_AUTHENTICTION	pp_router_mr_v1.1 pp_vpn_mr_v1.2	Reproduced exactly from the PPs
O.RESIDUAL_INFORMATION	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
O.RESOURCE_SHARING	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
O.THOROUGH_FUNCTIONAL_ TESTING	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
O.MEDIATE_INFORMATION_FLOW	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	This objective is identified as O.MEDIATE in pp_vpn_mr_v1.2 and pp_fw_tf_mr_v1.1. Reproduced exactly from the PPs
O.PROTOCOLS	pp_router_mr_v1.1	Reproduced exactly from the PP
O.PROTECT_IN_TRANSIT	pp_router_mr_v1.1	Reproduced exactly from the PP
O.REPLAY_DETECTION	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	This objective is identified as O.MEDIATE in pp_vpn_mr_v1.2 and pp_fw_tf_mr_v1.1. Reproduced exactly from the PPs
O.HA	New objective	This Objective was introduced in the ST. The objective does not contradict any of the functionality found in the MR PPs. This objective discusses High Availability objectives of the TOE.

2.3.4 Statement of Security Requirements Consistency

The Security Functional Requirements included in the Security Target represent a combination of the Security Functional Requirements specified in the three Protection Profiles for which conformance is claimed. All concepts covered in each of the Protection Profile's Statement of Security Requirements are included in the Security Target. Additionally, the Security Assurance Requirements included in the Security Target are identical to the Security Assurance Requirements included in each of the Protection Profiles. The following table identifies each SFR included in the ST and provides rationale for its inclusion in the Security Target with regards to the claims Protection Profiles.

Table 15: SFR Protection Profile Conformance

SFR	PPs	Rationale
FAU_ARP.1	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	This SFR is an exact reproduction of the SFR variants in pp_vpn_mr_v1.2 and pp_fw_tf_mr_v1.1 which include a superset of the SFR in pp_router_mr_v1.1.
FAU_GEN.1	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Identified as FAU_GEN.1-NIAP-0407 in pp_vpn_mr_v1.2 Identified as FAU_GEN.1-NIAP-0410 in pp_fw_tf_mr_v1.1 Identified as FAU_GEN.1-NIAP-0429 in pp_router_mr_v1.1 Since the SFRs present the same information in each iteration, the ST author chose the pp_router_mr_v1.1 to reproduce for readability. The ST includes all of the auditable events listed in each of the tables within each PP with the exception of events corresponding to the Authenticated Information Flow SFP.
FAU_GEN.2	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Identified as FAU_GEN.2-NIAP-410 in the PPs Exactly reproduced the version of the SFR included in pp_vpn_mr_v1.2 and pp_fw_tf_mr_v1.1 since these also include the events in the pp_router_mr_v1.1.
FAU_SAA.1	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Identified as FAU_SAA.1-NIAP-0407 in pp_vpn_mr_v1.2 and pp_fw_tf_mr_v1.1 Identified as FAU_SAA.1-NIAP-0410 in pp_router_mr_v1.1 Since the SFR presents the same information in each iteration with slight wording differences, the ST author chose to reproduce the variant of the SFR found in pp_fw_tf_mr_v1.1. The list of auditable events is a superset of the lists found in all of the SFRs in each PP.
FAU_SAR.1	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Exactly reproduced the variant of the SFR found in pp_vpn_mr_v1.2 and pp_fw_tf_mr_v1.1. These SFR variants were chosen because they identify the "administrators" as the role that may view the audit information. This is consistent with guidance provided by CCEVS.
FAU_SAR.2	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	These SFR variants were chosen because they identify the "administrators" as the role that may view the audit information. This is consistent with guidance provided by CCEVS.
FAU_SAR.3	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Exactly reproduced the variant of the SFR found in pp_vpn_mr_v1.2 and pp_fw_tf_mr_v1.1. These SFR variants were chosen because they identify all of the attributes listed in the SFR variants found in pp_vpn_mr_v1.2 plus "rule identity."
FAU_SEL.1	pp_router_mr_v1.1	Identified as FAU_SEL.1-NIAP-407 in the PPs, reproduced

SFR	PPs	Rationale
	pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	exactly from the PPs. The operation included in the SFR are a superset of the operations found in the SFR variants.
FAU_STG.1	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Identified as FAU_STG.1-NIAP-0429 in pp_router_mr_v1.1 and pp_vpn_mr_v1.2. Reproduced the SFRs exactly from the pp_router_mr_v1.1 and pp_fw_tf_mr_v1.1. These are not limited to "Unauthorized" modifications as identified in FAU_STG.1.2 of pp_vpn_mr_v1.2.
FAU_STG.3	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FCS_CKM.1(1)	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FCS_CKM.1(2)	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FCS_CKM.2	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FCS_CKM.4	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FCS_COP.1(1)	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FCS_COP.1(2)	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FCS_COP.1(3)	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from pp_vpn_mr_v1.2. The latest version of the VPN PP allows for SHA-1 as a selection. The reason is backward compatibility with IPSEC. The TOE implements IPSEC so this is a valid rationale for the use of SHA-1
FCS_COP.1(4)	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FDP_IFC.1(1)	pp_vpn_mr_v1.2	Reproduced exactly from the PP
FDP_IFC.1(2)	pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FDP_IFC.1(3)	pp_router_mr_v1.1	Not included in the ST. All requirements on Authenticated Information Flow SFP were excluded.
FDP_IFC.1(4)	pp_router_mr_v1.1 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FDP_IFF.1(1)	pp_vpn_mr_v1.2	Reproduced exactly from the PP
FDP_IFF.1(2)	pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FDP_IFF.1(3)	pp_router_mr_v1.1	Not included in the ST. All requirements on Authenticated Information Flow SFP were excluded.
FDP_IFF.1(4)	pp_router_mr_v1.1 pp_fw_tf_mr_v1.1	Used the variant found in pp_fw_tf_mr_v1.1 for the first element since it covers everything in pp_router_mr_v1.1 plus additional. Since the second element of both variants of the SFR contain

SFR	PPs	Rationale
		the same information worded slightly differently, the pp_fw_tf_mr_v1.1 variant was used for readability. Since the operation found in the sixth element of the SFR is more broadly stated in pp_fw_tf_mr_v1.1, this was used in the ST. All other text was replicated exactly from the PPs.
FDP_RIP.2	pp_router_mr_v1.1	Reproduced exactly from the PPs
FIA_AFL.1	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Identified as FIA_AFL.1-NIAP-0425 in pp_router_mr_v1.1. Reproduced the first assignment from pp_router_mr_v1.1 and pp_fw_tf_mr_v1.1 because it is more restrictive. Used the second operation from pp_vpn_mr_v1.2 and pp_fw_tf_mr_v1.1becasue it more specifically identifies the users of the TOE. All else reproduced exactly from the PPs. Note that for VPN peers utilizing IKEv1, as required in the PPs, it is not possible to support automated authentication failures. The IKEv1 protocol provides a pre-shared key method of an ISAKMP SA establishment, and when this method is used any IKE peer which possesses a pre-shared secret key is considered legitimate due to the anonymous nature of the IKEv1 DH key exchange procedure. Thus, policy based VPN peer lockout can only be achieved by manual methods (e.g. a pre-shared key removal or modification).
FIA_ATD.1(1)	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced the variant in pp_vpn_mr_v1.2. This variant identifies “administrator” rather then “authorized user”. This decision was made because the requirement more specific.
FIA_ATD.1(2)	pp_router_mr_v1.1	Reproduced exactly from the PP
FIA_UAU.1	pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FIA_UAU.2	pp_router_mr_v1.1	Reproduced exactly from the PPs
FIA_UID.2	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FIA_USB.1	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced the version of the SFR found in pp_router_mr_v1.1. This is because this version of the SFR was modified as per CCIMB Interp #137.
FMT_MOF.1(1)	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FMT_MOF.1(2)	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced the variant found in pp_vpn_mr_v1.2. This was done because this version of the SFR includes the names of the referenced SFRs. Removed the ability to disable selftests because it violates the rules of FIPS 140-2.
FMT_MOF.1(3)	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Exactly reproduced the variant of the SFR found in pp_vpn_mr_v1.2 and pp_fw_tf_mr_v1.1. These SFR variants were chosen because they identify the “administrators” as the role that may view the audit information. This is consistent with guidance provided by CCEVS.
FMT_MOF.1(4)	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FMT_MOF.1(5)	pp_router_mr_v1.1	Reproduced exactly from the PPs

SFR	PPs	Rationale
	pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	
FMT_MOF.1(6)	pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FMT_MOF.1(7)	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FMT_MOF.1(8)	pp_router_mr_v1.1	Reproduced exactly from the PP
FMT_MSA.1(1)	pp_fw_tf_mr_v1.1	Reproduced exactly from the PP
FMT_MSA.1(2)	pp_vpn_mr_v1.2	Refined the policy to only identify that the cryptographic administrator could update the policy attributes. This is consistent with the other Protection Profiles for which conformance is claimed and is consistent with guidance previously provided by CCEVS
FMT_MSA.1(3)	pp_router_mr_v1.1	Reproduced exactly from the PP
FMT_MSA.1(4)	pp_router_mr_v1.1	Not included in the ST. All requirements on Authenticated Information Flow SFP were excluded.
FMT_MSA.2	pp_router_mr_v1.1	Removed from the ST because this dependency is satisfied by placing strict requirements on the values of attributes of the cryptographic module in the associated FCS requirements. Therefore, FMT_MSA.2 is not necessary to satisfy the requirement of only secure values being assigned to secure attributes. This is consistent with pp_vpn_mr_v1.2 and pp_fw_tf_mr_v1.1. Since this requirement applies cryptographic parameters and pp_vpn_mr_v1 is the newest and most cryptographically intensive of the PPs for which conformance is claimed, the removal of FMT_MSA.2 is appropriate.
FMT_MSA.3(1)	pp_vpn_mr_v1.2	Reproduced exactly from the PP
FMT_MSA.3(2)	pp_vpn_mr_v1.2	Reproduced exactly from the PP
FMT_MSA.3(3)	pp_router_mr_v1.1 pp_fw_tf_mr_v1.1	Identified as FMT_MSA.3-NIAP-0409 in the pp_fw_tf_mr_v1.1 Combined the first operation of the SFRs to include the superset of the two operations. Used the refinement found in pp_fw_tf_mr_v1.1 for readability. This does not change the meaning of the SFR.
FMT_MTD.1(1)	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs Added “and the deletion of audit data” for consistency with the other SFRs in the PPs. This does not change the scope of the SFR. Removed “or authorized IT entities” since there are no IT entities that may change any of the security value. All administration is done by the administrator.
FMT_MTD.1(2)	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FMT_MTD.1(3)	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs Removed “or authorized IT entity” since time cannot be set from an external server. All administration is done by the administrator.
FMT_MTD.1(4)	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Combined the second operation of the SFR to include the superset of the operations of the variants of the SFR. No functionality was removed. This is consistent with guidance previously provided by

SFR	PPs	Rationale
		CCEVS.
FMT_MTD.2(1)	pp_fw_tf_mr_v1.1	Reproduced exactly from the PP
FMT_MTD.2(2)	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PP
FMT_MTD.2(3)	pp_router_mr_v1.1	Reproduced exactly from the PP
FMT_REV.1	pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	This includes the superset of the completion of the operations in the first and second element of the SFR. This is consistent with the other Protection Profiles for which conformance is claimed.
FMT_SMF.1	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	The operation in this SFR is the superset of the operation in each of the PPs. The first bullet was refined to remove the ability to invoke the TSF sel-test to the security administrator, which comflicted with FMT_SMR.2. The second bullet was refined to remove the ability to disable the crypto self tests, as the TOE does not support disabling them. The third bullet was refined to apply to administrators instead of just audit administrators so that it matches FMT_SMR.2 where all administrators may view audit records. Bullets 10, 12, and 14 were refined to remove the authorized IT entities role as NTP is not to be used with the TOE. The 23rd bullet came from the VPN PP but duplicates functionality from the Router PP in bullet 6.
FMT_SMR.2	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced the variant of the SFR found in pp_vpn_mr_v1.2. This variation of the SFR includes the entire scope of the SFR covered by the other variations
FPT_FLS.1	pp_router_mr_v1.1	Reproduced exactly from the PP
FPT_ITA.1	pp_router_mr_v1.1	Reproduced exactly from the PP
FPT_ITC.1	pp_router_mr_v1.1	Reproduced exactly from the PP
FPT_ITI.1	pp_router_mr_v1.1	Reproduced exactly from the PP
FPT_RCV.1	pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FPT_RCV.2	pp_router_mr_v1.1	Reproduced exactly from the PP
FPT_RPL.1	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FPT_STM.1	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FPT_TDC.1	pp_router_mr_v1.1	Reproduced exactly from the PP
FPT_TST.1(1)	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FPT_TST.1(2)	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FRU_RSA.1(1)	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Since the SFRs relate the same functionality the variant from pp_fw_tf_mr_v1.1 was used for readability. The functionality has not been affected.
FRU_RSA.1(2)	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FTA_SSL.1	pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Since the two variations of the SFRs within the present the same information with slightly different verbiage, the

SFR	PPs	Rationale
		variation found in pp_fw_tf_mr_v1.1 was used for readability.
FTA_SSL.2	pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Since the two variations of the SFRs within the present the same information with slightly different verbiage, the variation found in pp_fw_tf_mr_v1.1 was used for readability.
FTA_SSL.3	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Blended the requirement to include the more broad general session being terminated. This covers all of the types of sessions listed in the variation of the SFRs.
FTA_TAB.1	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Combined the first refinement in the SFR to include the both "user" and "administrator". Otherwise, the SFR is repeated exactly from the PPs.
FTA_TSE.1	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Since each variant of the SFR presents the same information in a slightly different manner, the variant found in pp_vpn_mr_v1.2 was replicated for readability.
FTP_ITC.1(1)	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FTP_ITC.1(2)	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs
FTP_TRP.1(1)	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Of the variations of this SFR, the one found in pp_fw_tf_mr_v1.1 covers the same information found in the other PPs plus additional information. The variant found in pp_fw_tf_mr_v1.1 was reproduced for clarity.
FTP_TRP.1(2)	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	The pp_router_mr_v1.1 variant of this SFR has the most comprehensive for the first element of the SFR. This version was used in the ST. The pp_router_mr_v1.1 and pp_fw_tf_mr_v1.1 variant of the second element are the broadest of the SFR variants. This version was used in the ST. The pp_router_mr_v1.1 and pp_fw_tf_mr_v1.1 variant of the third element are the broadest of the SFR variants. This version was used in the ST.
FAU_ARP_ACK_(EXT).1	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	The version of the SFR found in pp_router_mr_v1.1 was used because it contains all for the requirements of the other variations plus additional text. This extended requirement was included in the ST because of its inclusion in Protection Profiles for which conformance is claimed.
FAU_STG_(EXT).4	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Identified as FAU_STG.NIAP-0414-1-NIAP-0429 in the PPs. Since the variations of the SFR present similar information, the variation from pp_vpn_mr_v1.2 and pp_fw_tf_mr_v1.1 was chosen. This extended requirement was included in the ST because of its inclusion in Protection Profiles for which conformance is claimed.
FCS_CKM_(EXT).2	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs This extended requirement was included in the ST because of its inclusion in Protection Profiles for which conformance is claimed.

SFR	PPs	Rationale
FCS_COP_(EXT).1	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs This extended requirement was included in the ST because of its inclusion in Protection Profiles for which conformance is claimed.
FCS_IKE_(EXT).1	pp_router_mr_v1.1 pp_vpn_mr_v1.2	Reproduced the variant from pp_vpn_mr_v1.2. This was done to reflect the updated version of the requirement found in version 1.2 of the SFR. This extended requirement was included in the ST because of its inclusion in Protection Profiles for which conformance is claimed.
FCS_GDOI_(EXT).1	New Requirement not included in any PP.	Not applicable. New requirement. This requirement extends the IPSec VPN functionality to include Group Domain of Interpretation. This functionality does not contradict any other functionality found in the PPs.
FIA_UAU_(EXT).2	pp_fw_tf_mr_v1.1	Removed because the functionality is covered by FIA_UAU.2.
FIA_UAU_(EXT).5	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs This extended requirement was included in the ST because of its inclusion in Protection Profiles for which conformance is claimed.
FCS_BCM_(EXT).1	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Removed the portion of the requirement the required section 2 of the of the FIPS 140 standard (Cryptographic Module Ports and Interfaces) to be tested to level 3. This was removed with permission from CCEVS. This extended requirement was included in the ST because of its inclusion in Protection Profiles for which conformance is claimed.
FPT_HA_(EXT).1	New Requirement not included in any PP.	Not applicable. New requirement. This requirement does not contradict any of the functionality found in any of the other SFRs. This requirement only address the hardware high availability available in configurations of the TOE with multiple ESPs and RPs.
FPT_PRO_(EXT).1	pp_router_mr_v1.1	Reproduced exactly from the PP This extended requirement was included in the ST because of its inclusion in Protection Profiles for which conformance is claimed.
FPT_TST_(EXT).1	pp_router_mr_v1.1 pp_vpn_mr_v1.2 pp_fw_tf_mr_v1.1	Reproduced exactly from the PPs This extended requirement was included in the ST because of its inclusion in Protection Profiles for which conformance is claimed.

3 SECURITY PROBLEM DEFINITION

This chapter identifies the following:

- Significant assumptions about the TOE's operational environment.
- IT related threats to the organization countered by the TOE.
- Environmental threats requiring controls to provide sufficient protection.
- Organizational security policies for the TOE as appropriate.

This document identifies assumptions as A.assumption with "assumption" specifying a unique name. Threats are identified as T.threat with "threat" specifying a unique name. Policies are identified as P.policy with "policy" specifying a unique name.

3.1 Assumptions

The specific conditions listed in the following subsections are assumed to exist in the TOE's IT environment. These assumptions include both practical realities in the development of the TOE security requirements and the essential environmental conditions on the use of the TOE.

Table 16 TOE Assumptions

Assumption Name	Assumption Definition
A.NO_GENERAL_PURPOSE	The Administrator ensures there are no general purpose computing or storage repository capabilities (e.g., compilers, editors, web servers, database servers or user applications) available on the TOE.
A.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the environment.
A.NO_TOE_BYPASS	Information cannot flow between external and internal networks located in different enclaves without passing through the TOE.
A.AVAILABILITY	Network resources shall be available to allow clients to satisfy mission requirements and to transmit information.

NOTE: The Assumptions included in this ST are drawn from the three Medium Robustness Protection Profiles for which conformance is claimed. Similar Assumptions or Assumptions with the same purpose have been combined where appropriate.

3.2 Threats

The following table lists the threats addressed by the TOE and the IT Environment. The assumed level of expertise of the attacker for all the threats identified below is unsophisticated.

Table 17 Threats

Threat Name	Threat Definition
T.ADMIN_ERROR	An administrator may incorrectly install or configure the TOE, or install a corrupted TOE resulting in ineffective security mechanisms.
T.UNAUTHORIZED_ACCESS	A user may gain access to user data/services (either on the TOE or by sending data through the TOE) for which they are not authorized according to the TOE security policy.
T.ADDRESS_MASQUERADE	A user on one interface may masquerade as a user on another interface

Threat Name	Threat Definition
	to circumvent the TOE policy.
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 compromising the cryptographic mechanisms and the data protected by those mechanisms.
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.
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.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 thereby causing potential security vulnerabilities.
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.UNAUTHORIZED_PEER	An unauthorized IT entity may attempt to establish a security association with the TOE.
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.
T.ADMIN_ROGUE	An administrator's intentions may become malicious resulting in user or TOE Security Functions (TSF) data being compromised.
T.MASQUERADE	A malicious user, process, or external IT entity may masquerade as an authorized entity in order to gain access to data or TOE resources.
T.RESOURCE_EXHAUSTION	A malicious process or user may block others from system resources (e.g., connection state tables, TCP connections) via a resource exhaustion denial of service attack.
T.SPOOFING	A malicious user, process, or external IT entity may misrepresent itself as the TOE to obtain identification and authentication data.
T.UNATTENDED_SESSION	A user may gain unauthorized access to an unattended session.
T.EAVESDROP	A malicious user or process may observe or modify user or TSF data transmitted between physically separated parts of the TOE.
T.NORECOVERY	A single fault within the TSF may result in the TOE becoming non-operational and prevent TSF functionality.

NOTE: The Threats included in this ST are drawn from the three Medium Robustness Protection Profiles for which conformance is claimed. Similar Threats/Threats with the same purpose have been combined where appropriate.

3.3 Organizational Security Policies

An organizational security policy is a set of rules, practices, and procedures imposed by an organization to address its security needs. The following table, Organizational Security Policies, identifies the organizational security policies

Table 18 Organizational Security Policies

Policy Name	Policy Definition
P.CRYPTOGRAPHIC_FUNCTIONS	The TOE shall provide cryptographic functions for its own use, including encryption/decryption and digital signature operations.
P.CRYPTOGRAPHY_VALIDATED	Where the TOE requires FIPS-approved security functions, only NIST FIPS validated cryptography (methods and implementations) are acceptable for key management (i.e., generation, access, distribution, destruction, handling, and storage of keys) and cryptographic services (i.e., encryption, decryption, signature, hashing, key distribution, and random number generation 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.
P.INTEGRITY	The TOE shall support the IETF Internet Protocol Security Encapsulating Security Payload (IPSEC ESP) as specified in RFC 2406. Sensitive information transmitted to a peer TOE shall apply integrity mechanisms as specified in Use of HMAC-SHA-1-96 within ESP and AH (RFC 2404).
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 TOE.
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 both locally and remotely through protected communications channels.
P.CRYPTOGRAPHY	The TOE shall use NIST FIPS validated cryptography as a baseline with additional NSA-approved methods for key management (i.e., generation, access, distribution, destruction, handling, and storage of keys), and for cryptographic operations (i.e., encryption, decryption, signature, hashing, key exchange, and random number generation services).
P.COMPATIBILITY	The TOE must meet Request for Comments (RFC) requirements for implemented protocols to facilitate inter-operation with other routers and network equipment using the same protocols.

NOTE: The OSPs included in this ST are drawn from the three Medium Robustness Protection Profiles for which conformance is claimed. Similar OSPs or OSPs with the same purpose have been combined where appropriate.

4 SECURITY OBJECTIVES

This Chapter identifies the security objectives of the TOE and the IT Environment. The security objectives identify the responsibilities of the TOE and the TOE's IT environment in meeting the security needs.

- This document identifies objectives of the TOE as O.objective with objective specifying a unique name. Objectives that apply to the IT environment are designated as OE.objective with objective specifying a unique name.

4.1 Security Objectives for the TOE

The following table, Security Objectives for the TOE, identifies the security objectives of the TOE. These security objectives reflect the stated intent to counter identified threats and/or comply with any security policies identified. An explanation of the relationship between the objectives and the threats/policies is provided in the rationale section of this document.

Table 19 Security Objectives for the TOE

TOE Security Obj.	TOE Security Objective Definition
O.ROBUST_ADMIN_GUIDANCE	The TOE will provide administrators with the necessary information for secure delivery and management.
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 (i.e., encryption/decryption and digital signature operations) to maintain the confidentiality and allow for detection of modification of TSF data that is transmitted between physically separated portions of the TOE, or stored outside the TOE.
O.CRYPTOGRAPHY_ VALIDATED	The TOE shall use NIST FIPS 140-2 validated cryptomodules for cryptographic services implementing FIPS-approved security functions and random number generation services used by cryptographic functions.
O.DISPLAY_BANNER	The TOE will display an advisory warning regarding use of the TOE.
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.

TOE Security Obj.	TOE Security Objective Definition
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 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.USER_GUIDANCE	The TOE will provide users with the information necessary to correctly use the security mechanisms.
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.ADMIN_ROLE	The TOE will provide administrator roles to isolate administrative actions, and to make the administrative functions available locally and remotely.
O.INTEGRITY	The TOE must be able to protect the integrity of data transmitted to a peer TOE via encryption and provide IPSec authentication for such data. Upon receipt of data from a peer TOE, the TOE must be able to decrypt the data and verify that the received data accurately represents the data that was originally transmitted.
O.MAINT_MODE	The TOE shall provide a mode from which recovery or initial startup procedures can be performed.
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.PEER_AUTHENTICTION	The TOE will authenticate each peer TOE that attempts to establish a security association with the TOE.
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 to the TOE).
O.THOROUGH_FUNCTIONAL_ TESTING	The TOE will undergo appropriate security functional testing that demonstrates the TSF satisfies the security functional requirements.
O.MEDIATE_INFORMATION_FLOW	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.PROTOCOLS	The TOE will ensure that standardized protocols are implemented in the TOE to RFC and/or Industry specifications to ensure interoperability.
O.PROTECT_IN_TRANSIT	The TSF shall protect TSF data when it is in transit between the TSF and another trusted IT entity.
O.REPLAY_DETECTION	The TOE will provide a means to detect and reject the replay of authentication data and other TSF data and security attributes.

TOE Security Obj.	TOE Security Objective Definition
O.HA	The TOE shall provide mechanisms that allow continued operation of the TSF when a single hardware or software failure occurs within the TSF.

NOTE: The TOE Objectives included in this ST are drawn from the three Medium Robustness Protection Profiles for which conformance is claimed. Similar TOE Objectives or TOE Objectives with the same purpose have been combined where appropriate.

4.2 Security Objectives for the Environment

The assumptions identified previously are incorporated as security objectives for the environment. They levy additional requirements on the environment, which are largely satisfied through procedural or administrative measures. The following table, Security Objectives for the Environment, identifies the security objectives for the environment.

Table 20 Security Objectives for the Environment

Environment Security Objective Name	IT Environment Security Objective Definition
OE.CRYPTANALYTIC	Cryptographic methods used in the IT environment shall be interoperable with the TOE, should be FIPS 140-2 validated 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_GENERAL_PURPOSE	The Administrator ensures there are no general-purpose computing or storage repository capabilities (e.g., compilers, editors, or user applications) available on the TOE.
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.
OE.AVAILABILITY	Network resources will be available to allow clients to satisfy mission requirements and to transmit information.

NOTE: The IT Environment Objectives included in this ST are drawn from the three Medium Robustness Protection Profiles for which conformance is claimed. Similar IT Environment Objectives or IT Environment Objectives with the same purpose have been combined where appropriate.

5 SECURITY REQUIREMENTS

This section identifies the Security Functional Requirements for the TOE and for the IT Environment. The Security Functional Requirements included in this section are derived verbatim from Part 2 of the *Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 3, dated: July 2009* and all National Information Assurance Partnership (NIAP) and international interpretations.

5.1 Conventions

The CC defines operations on Security Functional Requirements: assignments, selections, assignments within selections and refinements. This document uses the following font conventions to identify the operations defined by the CC:

- Assignment: Indicated with *italicized* text;
- Refinement made by PP author: Indicated with **bold** text and strikethroughs, if necessary;
- Refinement made by ST author: Indicated with ***bold italicized*** text and strikethroughs, if necessary;
- Selection: Indicated with underlined text;
- Assignment within a Selection: Indicated with *italicized and underlined* text;
- Iteration: Indicated by appending the iteration number in parenthesis, e.g., (1), (2), (3).

Explicitly stated SFRs are identified by having a label '(EXT)' after the requirement name for TOE SFRs.

5.2 TOE Security Functional Requirements

This section identifies the Security Functional Requirements for the TOE. The TOE Security Functional Requirements that appear in the following table are described in more detail in the following subsections.

Table 21 Security Functional Requirements

Functional Component	
SFR Component ID	**Component Name**
Security Functional Requirements Directly Drawn from CC Part 2	
FAU_ARP.1	Security alarms
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_SEL.1	Selective Audit
FAU_STG.1	Protected audit trail storage
FAU_STG.3	Action in case of possible audit data loss
FCS_CKM.1(1)	Cryptographic Key Generation
FCS_CKM.1(2)	Cryptographic Key Generation
FCS_CKM.2	Cryptographic Key Distribution
FCS_CKM.4	Cryptographic Key Destruction
FCS_COP.1(1)	Cryptographic Operation

Functional Component	
FCS_COP.1(2)	Cryptographic Operation
FCS_COP.1(3)	Cryptographic Operation
FCS_COP.1(4)	Cryptographic Operation
FDP_IFC.1(1)	Subset information flow control
FDP_IFC.1(2)	Subset information flow control
FDP_IFC.1(4)	Subset information flow control
FDP_IFF.1(1)	Simple security attributes
FDP_IFF.1(2)	Simple security attributes
FDP_IFF.1(4)	Simple security attributes
FDP_RIP.2	Full residual information protection
FIA_AFL.1	Authentication failure handling
FIA_ATD.1(1)	User attribute definition
FIA_ATD.1(2)	User attribute definition
FIA_UAU.1	Timing of authentication
FIA_UAU.2	User authentication before any action
FIA_UID.2	User identification before any action
FIA_USB.1	User-Subject Binding
FMT_MOF.1(1)	Management of security functions behavior
FMT_MOF.1(2)	Management of security functions behavior
FMT_MOF.1(3)	Management of security functions behavior
FMT_MOF.1(4)	Management of security functions behavior
FMT_MOF.1(5)	Management of security functions behavior
FMT_MOF.1(6)	Management of security functions behavior
FMT_MOF.1(7)	Management of security functions behavior
FMT_MOF.1(8)	Management of security functions behavior
FMT_MSA.1(1)	Management of security attributes
FMT_MSA.1(2)	Management of security attributes
FMT_MSA.1(3)	Management of security attributes
FMT_MSA.3(1)	Static attribute initialization
FMT_MSA.3(2)	Static attribute initialization
FMT_MSA.3(3)	Static attribute initialization
FMT_MTD.1(1)	Management of TSF data
FMT_MTD.1(2)	Management of TSF data
FMT_MTD.1(3)	Management of TSF data
FMT_MTD.1(4)	Management of TSF data
FMT_MTD.2(1)	Management of limits on TSF data
FMT_MTD.2(2)	Management of limits on TSF data
FMT_MTD.2(3)	Management of limits on TSF data
FMT_REV.1	Revocation
FMT_SMF.1	Specification of Management Functions
FMT_SMR.2	Restrictions on security roles
FPT_FLS.1	Failure with preservation of secure state
FPT_ITA.1	Inter-TSF availability within a defined availability metric
FPT_ITC.1	Inter-TSF confidentiality during transmission
FPT_ITI.1	Inter-TSF detection of modification
FPT_RCV.1	Manual recovery
FPT_RCV.2	Automated Recovery
FPT_RPL.1	Replay detection
FPT_STM.1	Reliable time stamps
FPT_TDC.1	Inter-TSF basic TSF data consistency

Functional Component	
FPT_TST.1(1)	TSF Testing
FPT_TST.1(2)	TSF Testing
FRU_RSA.1(1)	Maximum quotas
FRU_RSA.1(2)	Maximum quotas
FTA_SSL.1	TSF-initiated session locking
FTA_SSL.2	User-initiated locking
FTA_SSL.3	TSF-initiated termination
FTA_TAB.1	Default TOE access banners
FTA_TSE.1	TOE session establishment
FTP_ITC.1(1)	Inter-TSF trusted channel
FTP_ITC.1(2)	Inter-TSF trusted channel
FTP_TRP.1(1)	Trusted path
FTP_TRP.1(2)	Trusted path
Explicitly Stated Security Functional Requirements	
FAU_ARP_ACK_(EXT).1	Security alarm acknowledgement
FAU_STG_(EXT).4	Site-Configurable Prevention of Audit Loss
FCS_CKM_(EXT).2	Explicit: Cryptographic Key Handling and Storage
FCS_COP_(EXT).1	Explicit: Random Number Generation
FCS_IKE_(EXT).1	Internet Key Exchange
FCS_GDOI_(EXT).1	Group Domain of Interpretation
FIA_UAU_(EXT).5	Multiple authentication mechanisms
FCS_BCM_(EXT).1	Explicit: Baseline Cryptographic Module
FPT_HA_(EXT).1	High Availability
FPT_PRO_(EXT).1	Standard Protocol Usage
FPT_TST_(EXT).1	Extended: TSF Testing

5.2.1 Security audit (FAU)

FAU_ARP.1 Security alarms

FAU_ARP.1.1 – **Refinement:** The TSF shall [*immediately display an alarm message, identifying the potential security violation and make accessible the audit record contents associated with the auditable event(s) that generated the alarm, at the:*

a) local console,
b) remote administrator sessions that exist, and;
c) remote administrator sessions that are initiated before the alarm has been acknowledged, and;
d) at the option of the Security Administrator, generate an audible alarm, and;
e) *[no other methods]]*

upon detection of a potential security violation.

FAU_GEN.1 Audit data generation

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

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

b) All auditable events for the *basic* level of audit;
c) [*specifically defined auditable events listed in Table 21;*] **and**
d) [no additional events].

FAU_GEN.1.2 **Refinement:** The TSF shall record within each audit record at least the following information:

a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event; and
b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [*information specified in column three* **of the table below**].

Table 22 Auditable Events Table

Requirement	Auditable Events	Audit Record Contents
FAU_ARP.1	Actions taken due to potential security violations.	Identification of what caused the generation of the alarm.
	Potential security violation was detected	
FAU_ARP_ACK_(EXT).1	Actions taken due to potential security violations.	The identity and location 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; Automated responses performed by the tool.	The identity of the Security Administrator performing the function.
FAU_SAR.1	Reading of information from the audit records.	The identity of the Audit Administrator performing the function.
	Opening the audit trail	
FAU_SAR.2	Unsuccessful attempts to read information from the audit records.	The identity of the administrator performing the function.
FAU_SAR.3	Unsuccessful attempts to read information from the audit records.	The parameters used for the viewing.
FAU_SEL.1	All modifications to the audit configuration that occur while the audit collection functions are operating.	The identity of the Security Administrator performing the function.
FAU_STG_(EXT).4	Actions taken due to the audit storage failure.	The identity of the Security Administrator performing the function.
FAU_STG.1	None	
FAU_STG.3	Actions taken due to exceeding the audit threshold.	The identity of the Security Administrator performing the function.
FCS_BCM_(EXT).1	None	
FCS_CKM.1(1)	a) Failure of the activity; b) Generation and loading of key.	Identify the failed activity and the data that caused the failure.
	Failure of the symmetric key generation	Type of cryptographic operation Any applicable cryptographic mode(s) of operation, excluding any sensitive information
FCS_CKM.1(2)	a) Failure of the activity; b) Generation and loading of key pair for digital signatures.	Identify the failed activity and the data that caused the failure.

Requirement	Auditable Events	Audit Record Contents
	Failure of the asymmetric key generation	Type of cryptographic operation Any applicable cryptographic mode(s) of operation, excluding any sensitive information
FCS_CKM.2	a) Failure of the activity; b) Generation and loading of key.	Identify the failed activity and the data that caused the failure.
	Failure of the Key distribution	Type of cryptographic operation Any applicable cryptographic mode(s) of operation, excluding any sensitive information
FCS_CKM.4	a) Failure of the activity; b) Generation and loading of key.	Identify the failed activity and the data that caused the failure.
	Failure of the Key destruction	Type of cryptographic operation Any applicable cryptographic mode(s) of operation, excluding any sensitive information
FCS_CKM_(EXT).2	a) Failure of the activity; b) Generation and loading of key.	Identify the failed activity and the data that caused the failure.
	Failure of the Cryptographic Key Handling and Storage	Type of cryptographic operation Any applicable cryptographic mode(s) of operation, excluding any sensitive information
FCS_COP.1(1)	Failure of cryptographic operation.	Type of cryptographic operation. Any applicable cryptographic mode(s) of operation, excluding any sensitive information.
FCS_COP.1(2)	Failure of cryptographic operation.	Type of cryptographic operation Any applicable cryptographic mode(s) of operation, excluding any sensitive information.
FCS_COP.1(3)	Failure of cryptographic operation.	Type of cryptographic operation. Any applicable cryptographic mode(s) of operation, excluding any sensitive information.
FCS_COP.1(4)	Failure of cryptographic operation.	Type of cryptographic operation. Any applicable cryptographic mode(s) of operation, excluding any sensitive information.
FCS_COP_(EXT).1	Failure of cryptographic operation.	Type of cryptographic operation. Any applicable cryptographic mode(s) of operation, excluding any sensitive information.
FCS_IKE_(EXT).1	a) Generation and loading of key pair for digital signatures; b) Changes to the pre-shared key used for authentication; c) All modifications to the key lifetimes; d) Failure of the authentication in Phase 1; e) Failure to negotiate a security association in Phase 2.	If failure occurs, record a descriptive reason for the failure.
FCS_GDOI_(EXT).1	Failure of cryptographic operation.	If failure occurs, record a descriptive reason for the failure.
FDP_IFC.1(1)	None	

Requirement	Auditable Events	Audit Record Contents
FDP_IFC.1(2)	None	
FDP_IFC.1(4)	None	
FDP_IFF.1(1)	a) Decisions to permit or deny information flows ***(This encompasses "all decisions on requests for information flow");*** b) Operation applied to each information flow permitted.	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 interface on which the TOE received the packet. For denied information flows, the reason for denial.
FDP_IFF.1(2)	Decisions to permit or deny information flows ***(This encompasses "all decisions on requests for information flow").***	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 interface on which the TOE received the packet. For denied information flows, the reason for denial.
FDP_IFF.1(4)	Decisions to permit or deny information flows ***(This encompasses "all decisions on requests for information flow").***	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 interface on which the TOE received the packet. For denied information flows, the reason for denial.
FDP_RIP.2	None	
FIA_AFL.1	a) The reaching of the threshold for the unsuccessful authentication attempts and the actions (e.g., disabling of an account) taken and the subsequent, if appropriate, restoration to the normal state (e.g., re-enabling of a terminal).	Identity of the unsuccessful authentication attempts, Terminal identification, Action taken.
FIA_ATD.1(1)	None	
FIA_ATD.1(2)	None	
FIA_UAU.1	None	
FIA_UAU.2	a) Successful and unsuccessful use of authentication mechanisms; b) All use of the authentication mechanism.	Claimed identity of the user using the authentication mechanism. Success or failure of the authentication mechanism.
FIA_UAU_(EXT).5	a) The final decision on authentication; b) The result of each activated mechanism together with the final decision. All use of the local authentication mechanism	Claimed identity of the user attempting to authenticate.

Requirement	Auditable Events	Audit Record Contents
FIA_UID.2	a) Unsuccessful use of the user identification mechanism, including the user identity provided; b) All use of the user identification mechanism, including the user identity provided (that is, those that authenticate to the TOE).	Claimed identity of the user using the identification mechanism.
	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	a) Unsuccessful binding of user security attributes to a subject (e.g., creation of a subject). b) 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(1)	All modifications in the behavior of the functions in the TSF.	The identity of the administrator performing the function, the function being performed and the data being used to perform the function (if available).
FMT_MOF.1(2)	a) Enabling of the key-generation self-tests. b) All modifications in the behavior of the functions in the TSF.	The identity of the administrator performing the function, the function being performed and the data being used to perform the function (if available).
FMT_MOF.1(3)	All modifications in the behavior of the functions in the TSF.	The identity of the administrator performing the function, the function being performed and the data being used to perform the function (if available).
FMT_MOF.1(4)	All modifications in the behavior of the functions in the TSF.	The identity of the administrator performing the function, the function being performed and the data being used to perform the function (if available).
FMT_MOF.1(5)	All modifications in the behavior of the functions in the TSF.	The identity of the administrator performing the function, the function being performed and the data being used to perform the function (if available).
FMT_MOF.1(6)	All modifications in the behavior of the functions in the TSF.	The identity of the administrator performing the function, the function being performed and the data being used to perform the function (if available).
FMT_MOF.1(7)	All modifications in the behavior of the functions in the TSF.	The identity of the administrator performing the function, the function being performed and the data being used to perform the function (if available).
FMT_MOF.1(8)	All modifications in the behavior of the functions in the TSF.	The identity of the administrator performing the function, the function being performed and the data being used to perform the function (if available).
FMT_MSA.1(1)	All manipulation of the security attributes. ***(This encompasses "All modifications of the values of security attributes.")***	The identity of the administrator performing the function, the function being performed and the security attributes being used to perform the function (if available).

Requirement	Auditable Events	Audit Record Contents
FMT_MSA.1(2)	All manipulation of the security attributes. ***(This encompasses "All modifications of the values of security attributes.")***	The identity of the administrator performing the function, the function being performed and the security attributes being used to perform the function (if available).
FMT_MSA.1(3)	All manipulation of the security attributes. ***(This encompasses "All modifications of the values of security attributes.")***	The identity of the administrator performing the function, the function being performed and the security attributes being used to perform the function (if available).
FMT_MSA.3(1)	a) Modifications of the default setting of permissive or restrictive rules; b) All modifications of the initial values of security attributes.	The identity of the administrator performing the function, the function being performed and the security attributes being used to perform the function (if available).
FMT_MSA.3(2)	a) Modifications of the default setting of permissive or restrictive rules; b) All modifications of the initial values of security attributes.	The identity of the administrator performing the function, the function being performed and the security attributes being used to perform the function (if available).
FMT_MSA.3(3)	a) Modifications of the default setting of permissive or restrictive rules; b) All modifications of the initial values of security attributes.	The identity of the administrator performing the function, the function being performed and the security attributes being used to perform the function (if available).
FMT_MTD.1(1)	All modifications of the values of TSF data by the administrator. ***(This encompasses "All modifications to the values of TSF data.")***	The identity of the administrator performing the function, the function being performed and the values of TSF data being modified during the performance the function (if available).
FMT_MTD.1(2)	All modifications of the values of cryptographic security data by the cryptographic administrator. ***(This encompasses "All modifications to the values of TSF data.")***	The identity of the administrator performing the function, the function being performed and the values of TSF data being modified during the performance the function (if available).
FMT_MTD.1(3)	All modifications to the time and date used to form the time stamps by the administrator. ***(This encompasses "All modifications to the values of TSF data.")***	The identity of the administrator performing the function, the function being performed the values of data being modified and the modifying data used during the performance the function.
FMT_MTD.1(4)	All modifications to the information flow policy ruleset by the Security Administrator. ***(This encompasses "All modifications to the values of TSF data.")***	The identity of the administrator performing the function, the function being performed the values of data being modified and the modifying data used during the performance the function.
FMT_MTD.2(1)	a) All modifications of the limits on TSF data b) All modifications in the actions to be taken in case of violation of the limits.	The identity of the administrator performing the function, the function being performed the values of data being modified and the modifying data used during the performance the function.

Requirement	Auditable Events	Audit Record Contents
	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_MTD.2(2)	a) All modifications of the limits on TSF data. b) All modifications in the actions to be taken in case of violation of the limits.	The identity of the administrator performing the function, the function being performed the values of data being modified and the modifying data used during the performance the function.
	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_MTD.2(3)	a) All modifications of the limits on TSF data. b) All modifications in the actions to be taken in case of violation of the limits.	The identity of the administrator performing the function, the function being performed the values of data being modified and the modifying data used during the performance the function.
FMT_REV.1	a) Unsuccessful revocation of security attributes; b) 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	Use of the management functions.	The identity of the administrator performing the function. Identify the management function being performed
FMT_SMR.2	a) Modifications to the group of users that are part of a role; b) Unsuccessful attempts to use a role due to given conditions on the roles.	User IDs which are associated with the modifications. The identity of the administrator performing the function.
FPT_FLS.1	Failure of the TSF.	Indication that the TSF has failed with the type of failure that occurred.
FPT_HA_(EXT).1	Failure of the TSF.	Indication that the TSF has failed with the type of failure that occurred.
FPT_ITA.1	The absence of TSF data when required by a TOE.	Include the type of TSF data that was not available and the condition that was to ensure availability.
FPT_ITC.1	None	
FPT_ITI.1	a) The detection of modification of transmitted TSF data. b) The action taken upon detection of modification of transmitted TSF data.	Identify the data that detected as modified and the action taken upon detection of the modification.
FPT_PRO_(EXT).1	None	
FPT_RCV.1	a) The fact that a failure or service discontinuity occurred; b) Resumption of the regular operation; c) Type of failure or service discontinuity.	Type of failure or service discontinuity
FPT_RCV.2	a) The fact that a failure or service discontinuity occurred; b) Resumption of the regular operation; c) Type of failure or service discontinuity.	Identify the failure, and that the TSF was able to recover to a secure state. If it is not possible to recover, enter maintenance mode.
FPT_RPL.1	Detected replay attacks.	Identity of the user that was the subject

Requirement	Auditable Events	Audit Record Contents
(including replay of authentication data notification from the authentication server)		of the reply attack
	Notification that a replay event occurred	Identity of the user that was the subject of the reply attack
FPT_STM.1	Changes to the time.	Identify that the time has been changed and the administrator that took the action.
		The identity of the administrator if the change was performed by an administrator
FPT_TST_(EXT).1	Execution of this set of TSF self tests and the results of the tests.	The identity of the administrator performing the test, if initiated by an administrator. Report any results from the test.
FPT_TST.1(1)	Execution of this set of TSF self tests for Cryptography and the results of the tests.	The identity of the administrator performing the test, if initiated by an administrator. Report any results from the test.
FPT_TST.1(2)	Execution of this set of TSF self tests for key generation and the results of the tests.	The identity of the administrator performing the test, if initiated by an administrator. Report any results from the test.
FRU_RSA.1(1)	a) Rejection of allocation operation due to resource limits. b) All attempted uses of the resource allocation functions for resources that are under control of the TSF.	Identify the controlled resources (transport-layer quotas) that caused the rejection, and the source subject identifier.
FRU_RSA.1(2)	a) Rejection of allocation operation due to resource limits. b) All attempted uses of the resource allocation functions for resources that are under control of the TSF.	Identify the controlled resources (controlled connection-oriented resources) that caused the rejection, and the user.
FTA_SSL.1	Locking of an interactive session by the session locking mechanism Successful unlocking of an interactive session. Any attempts at unlocking of an interactive session	The identity of the user associated with the session being locked or unlocked
FTA_SSL.2	a) Locking of an interactive session by the session locking mechanism. b) Successful unlocking of an interactive session. c) Any attempts at unlocking an interactive session.	The identity of the user associated with the session being locked or unlocked
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	a) Denial of a session establishment due to the session establishment mechanism. b) 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.

Requirement	Auditable Events	Audit Record Contents
FTP_ITC.1(1)	a) Failure of the trusted channel functions. b) Identification of the initiator and target of failed trusted channel functions. c) All attempted uses of the trusted channel functions. d) Identifier of the initiator and target of all trusted channel functions.	Indicated that the trusted channel failed and identification of the initiator and target of all trusted channels.
FTP_ITC.1(2)	a) Failure of the trusted channel functions. b) Identification of the initiator and target of failed trusted channel functions. c) All attempted uses of the trusted channel functions. d) Identifier of the initiator and target of all trusted channel functions.	Indicated that the trusted channel failed and identification of the initiator and target of all trusted channels.
FTP_TRP.1(1)	a) Failures of the trusted path functions. b) Identification of the user associated with all trusted path failures, if available. c) All attempted uses of the trusted path functions. d) Identification of the user associated with all trusted path invocations, if available.	Indicated that the trusted path failed and Identification of the claimed user identity.
FTP_TRP.1(2)	a) Failures of the trusted path functions. b) Identification of the user associated with all trusted path failures, if available. c) All attempted uses of the trusted path functions. d) Identification of the user associated with all trusted path invocations, if available.	Indicated that the trusted channel failed and Identification of the claimed user identity.

FAU_GEN.2 User identity association

FAU_GEN.2.1 – **Refinement**: The TSF shall be able to associate each auditable event with the identity of the user that caused the event.

FAU_SAA.1 Potential violation analysis

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

FAU_SAA.1.2 **- Refinement:** The TSF shall enforce the following rules for monitoring audited events:

a) [*Security Administrator specified number of authentication failures;*
b) Security Administrator specified number of Information Flow policy violations by an individual presumed source network identifier (e.g., IP address) within an administrator specified time period;
c) Security Administrator specified number of Information Flow policy violations to an individual destination network identifier within an administrator specified time period;

d) Security Administrator specified number of Information Flow policy violations to an individual destination subject service identifier (e.g., TCP port) within an administrator specified time period;
e) Security Administrator specified Information Flow policy rule, or group of rule violations within an administrator specified time period;
f) Any detected replay of TSF data or security attributes;
g) Any failure of the cryptomodule/cryptographic self-tests (FPT_TST.1(1));
h) Any failure of the other key generation self-tests (FPT_TST.1(2));
i) Any failure of the other TSF self-tests (FPT_TST_(EXT).1);
j) Security Administrator specified number of encryption failures;
k) Security Administrator specified number of decryption failures;
l) Security Administrator specified number of Phase 1 authentication failures when negotiating the Internet Key Exchange protocol;
m) Security Administrator specified number of failures occur during Phase 2 negotiation; and
n) [no additional rules]

known to indicate a potential security violation;

FAU_SAR.1 Audit review

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

FAU_SAR.1.2 – **Refinement**: The TSF shall provide the audit records in a manner suitable for the **Administrators** to interpret the information.

FAU_SAR.2 Restricted audit review

FAU_SAR.2.1 – **Refinement**: The TSF shall prohibit all users read access to the audit records **in the audit trail**, except **the Administrators**.

FAU_SAR.3 Selectable audit review

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

a) [*user identity;*
b) source subject identity;
c) destination subject identity;
d) ranges of one or more: dates, times, user identities, subject service identifiers, or transport layer protocol;
e) Rule identity;
f) TOE network interfaces; and
g) [no additional criteria].

FAU_SEL.1 Selective Audit

FAU_SEL.1.1 - **Refinement**: The TSF shall **allow only the Security Administrator** to include or exclude auditable events from the set of audited events based on the following attributes:

a) user identity;
b) event type;
c) [none];
d) [*network identifier;*
e) subject service identifier;
f) success of auditable security events;
g) failure of auditable security events;
h) rule identity; and
i) [no additional criteria].

FAU_STG.1 Protected audit trail storage

FAU_STG.1.1 – **Refinement**: The TSF shall **restrict the deletion of** stored audit records in the audit trail **to the Audit Administrator**.

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

FAU_STG.3 Action in case of possible audit data loss

FAU_STG.3.1 - **Refinement**: The TSF shall *immediately alert the administrators by displaying a message at the local console, and at the remote administrative console when an administrative session exists for each of the defined administrative roles, at the option of the Security Administrator generate an audible alarm,* [no other methods] if the audit trail exceeds [*a Security Administrator settable percentage of storage capacity*].

5.2.2 Cryptographic Support (FCS)

FCS_CKM.1(1) Cryptographic Key Generation (for symmetric keys)

FCS_CKM.1.1(1) **Refinement:** The TSF shall generate symmetric cryptographic keys **using a FIPS-Approved Random Number Generator as specified in FCS_COP_(EXT).1, and provide integrity protection to generated symmetric keys in accordance with NIST SP 800-57 "Recommendation for Key Management" Section 6.1.**

FCS_CKM.1(2) Cryptographic Key Generation (for asymmetric keys)

FCS_CKM.1.1(2) **Refinement:** The TSF shall generate **asymmetric** cryptographic keys in accordance **with the mathematical specifications of the FIPS-approved or NIST-recommended standard** [*ANSI X9.31*], **using a domain parameter generator and** [

1. *a FIPS-Approved Random Number Generator as specified in FCS_COP_(EXT)*]

in a cryptographic key generation scheme that meets the following:

- The TSF shall provide integrity protection and assurance of domain parameter and public key validity to generated asymmetric keys in accordance with NIST SP 800-57 "Recommendation for Key Management" Section 6.1.
- Generated key strength shall be equivalent to, or greater than, a symmetric key strength of 128 bits using conservative estimates.

FCS_CKM.2 Cryptographic Key Distribution

FCS_CKM.2.1 The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method [

1. *Manual (Physical) Method, and/or*
2. *Automated (Electronic) Method*]

that meets the following:

- *NIST Special Publication 800-57, "Recommendation for Key Management" Section 8.1.5*
- *NIST Special Publication 800-56A, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography"*

FCS_CKM.4 Cryptographic Key Destruction

FCS_CKM.4.1 **Refinement:** The TSF shall destroy cryptographic keys in accordance with a **cryptographic key zeroization method** that meets the following:

a) Key zeroization requirements of FIPS PUB 140-2, "Security Requirements for Cryptographic Modules"
b) Zeroization of all plaintext cryptographic keys and all other critical cryptographic security parameters shall be immediate and complete.
c) The TSF shall zeroize each intermediate storage area for plaintext key/critical cryptographic security parameter (i.e., any storage, such as memory buffers, that is included in the path of such data) upon the transfer of the key/critical cryptographic security parameter to another location.
d) For non-volatile memories other than EEPROM and Flash, the zeroization shall be executed by overwriting three or more times using a different alternating data pattern each time.
e) For volatile memory and non-volatile EEPROM and Flash memories, the zeroization shall be executed by a single direct overwrite consisting of a pseudo random pattern, followed by a read-verify.

FCS_COP.1(1) Cryptographic Operation (for data encryption/decryption)

FCS_COP.1.1(1) **Refinement:** The cryptomodule shall perform **encryption and decryption using the FIPS-approved security function AES algorithm operating in** [*CBC mode*] **and cryptographic key size of** [128 bits, 192 bits, 256 bits].

FCS_COP.1(2) Cryptographic Operation (for cryptographic signature)

FCS_COP.1.1(2) **Refinement:** The TSF shall perform **cryptographic signature services using the FIPS-approved security function** [RSA Digital Signature Algorithm (rDSA) with a key size (modulus) of [*between 3072 bits and 4096 bits*]]

that meets NIST Special Publication 800-57, "Recommendation for Key Management."

FCS_COP.1(3) Cryptographic Operation (for cryptographic hashing)

FCS_COP.1.1(3) **Refinement:** The TSF shall perform **cryptographic hashing services using the FIPS-approved security function Secure Hash Algorithm and** any message digest specified in FIPS 180-2 [SHA-1].

FCS_COP.1(4) Cryptographic Operation (for cryptographic key agreement)

FCS_COP.1.1(4) **Refinement:** The TSF shall perform **cryptographic key agreement services using the FIPS-approved security function as specified in NIST Special Publication 800-56A, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography"** [[*Finite Field-based key agreement algorithm*] **and cryptographic key sizes (modulus) of** [*3072 bits and 4096 bits*]] that meets NIST Special Publication 800-57, "Recommendation for Key Management".

5.2.3 User data protection (FDP)

FDP_IFC.1(1) Subset information flow control (VPN policy)

FDP_IFC.1.1(1) - The TSF shall enforce the [*VPN SFP*] on

a) [*source subject: TOE interface on which information is received;*
b) *destination subject: TOE interface to which information is destined.*
c) *information: network packets; and*
d) *operations:*

i. *pass packets without modifying;*
ii. *send IPSec encrypted and authenticated packets to a peer TOE using ESP in tunnel mode as defined in RFC 2406;*
iii. *decrypt, verify authentication and pass received packets from a peer TOE in tunnel mode using ESP;*
iv. [*none*]].

FDP_IFC.1(2) Subset information flow control (unauthenticated TOE services policy)

FDP_IFC.1.1(2) - The TSF shall enforce the [*UNAUTHENTICATED TOE SERVICES SFP*] on

a) [*source subject: TOE interface on which information is received;*
b) *destination subject: the TOE;*

c) information: network packets; and
d) *operations: accept or reject network packet*].

FDP_IFC.1(4)[1] Subset information flow control (unauthenticated policy)

FDP_IFC.1.1(4) - The TSF shall enforce the [*UNAUTHENTICATED INFORMATION FLOW SFP*] on [

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

    1) pass information by opening a relay connection through the TSF on behalf of the source subject to the destination subject, and with the TSF ensuring the following conditions:

        a) the connection from the source subject is from a valid peer network,
        b) the new relay connection is established to the destination subject on a valid peer network.]

    2) *pass information.*

FDP_IFF.1(1) Simple security attributes (VPN policy)

FDP_IFF.1.1(1) - The TSF shall enforce the [*VPN SFP*] based on the following types of subject and information security attributes:

a) [*Source subject security attributes:*
- *set of source subject identifiers **(IP address)**; and*
- [none].

b) Destination subject security attributes:
- *Set of destination subject identifiers **(IP address)**; and*
- [none].

c) Information security attributes:
- *presumed identity of source subject[2];*
- *identity of destination subject;*

FDP_IFF.1.2(1) - **Refinement**: The TSF shall permit an information flow between a **source subject and a destination subject** via a controlled operation if the following rules hold:

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

[1] The third iteration of FDP_IFC.1 was removed from the ST. Iteration numbering was preserved for document compatibility.

- *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 [The TOE examines a packet's source IP address, destination IP address, transport protocol, and layer 4 source and destination ports and compares them to the configured VPN policy to determine the action to apply to the network packets. If the packet is a plaintext packet that matches a policy rule that allows packets to be passed without modification, the packet is passed without modification. If the packet is a plaintext packet that matches a policy rule that requires the TOE to send IPSec encrypted and authenticated packets to a peer, the TOE encrypts and applies an authentication mechanism to the packet using ESP in tunnel mode as defined in RFC 2406 and sends it to its peer. If the packet matches a policy that requires the TOE to decrypt, verify authentication and pass received packets from a peer TOE in tunnel mode using ESP, the TOE decrypts, verifies authentication and passes received packets from a peer TOE in tunnel mode using ESP]; and*
- *the selected information flow policy rule specifies that the information flow is to be permitted, and what specific operation from FDP_IFC.1(1) is to be applied to that information flow***]**.

FDP_IFF.1.3(1) - The TSF shall enforce the [*none*]

FDP_IFF.1.4(1) - The TSF shall provide the following [*the Security Administrator shall have the capability to view all information flows allowed by the information flow policy ruleset before the ruleset is applied*].

FDP_IFF.1.5(1) - The TSF shall explicitly authorize an information flow based on the following rules: [*none*].

FDP_IFF.1.6(1) - The TSF shall explicitly deny an information flow based on the following rules:

a) [*The TOE shall reject requests for access or services where the presumed source identity of the information received by the TOE is not included in the set of source identifiers for the source subject;*
b) The TOE shall reject requests for access or services where the presumed source identity of the information received by the TOE specifies a broadcast identity;
c) The TOE shall reject requests for access or services where the presumed source identity of the information received by the TOE specifies a loopback identifier;
d) *The TOE shall reject requests in which the information received by the TOE contains the route (set of host network identifiers) by which information shall flow from the source subject to the destination subject.)*].

FDP_IFF.1(2) Simple security attributes (unauthenticated TOE services policy)

FDP_IFF.1.1(2) - The TSF shall enforce the [*UNAUTHENTICATED TOE SERVICES SFP*] based on the following types of subject and information security attributes:

a) [*Source subject security attributes:*
- *set of source subject identifiers **(IP address)**; and*
- [none].

b) Destination subject security attributes:
- *TOE's network identifier **(IP address)**; and*
- [none].

c) Information security attributes:
- *presumed identity of source subject;*
- *identity of destination subject;*
- *transport layer protocol;*
- *source subject service identifier;*
- *destination subject service identifier (e.g., TCP or UDP destination port number); and*
- [for an IP-based network stack:
- ICMP message type and code as specified in RFC 792, [[
  - *presumed source IP address*
  - *destination IP address*
  - none]];

  or for a non-IP-based network stack: [
  - *none*]].

FDP_IFF.1.2(2) – **Refinement**: The TSF shall permit an information flow between a **source** subject and **the TOE** via a controlled operation if the following rules hold:

- [*the presumed identity of the source subject is in the set of source subject identifiers;*
- *the identity of the destination subject is the TOE;*
- *the information security attributes match the attributes in an information flow control policy according to the following algorithm* [
  - *Network traffic is received by the TOE on one of its interfaces. These interfaces are grouped into zones. Traffic destined for the TOE itself is destined for the "self" zone,*
  - *The TOE examines the packet security attributes (including, presumed identity of source subject (IP address), identity of destination subject (IP address), transport layer protocol, source subject service identifier, destination subject service identifier (e.g., TCP or UDP destination port number) and compares the packet to the configured information flow policy rules,*
  - *The TOE identifies the configured information flow rule for which the network traffic applies. The information flow rules identify the actions allowed between the configured ingress zone of interfaces and the TOE "self" zone,*
- *The TOE accepts the network traffic if it meets a configured allow policy and does not meet a configured reject policy*].

FDP_IFF.1.3(2) - The TSF shall enforce the [*following rules:*

- *The TOE shall allow source subjects to access TOE services* [for an IP-based network stack: ICMP, [[*none*]; or for non-IP-based network stacks: *[none*] without authenticating those source subjects; and
- The TOE shall allow the list of services specified immediately above to be enabled (become available to unauthenticated users) or disabled (become unavailable to unauthenticated users)].

FDP_IFF.1.4(2) - The TSF shall provide the following [*the Security Administrator shall have the capability to view all information flows allowed by this information flow control policy before the policy is applied*].

FDP_IFF.1.5(2) - The TSF shall explicitly authorize an information flow based on the following rules: [*none*].

FDP_IFF.1.6(2) - The TSF shall explicitly deny an information flow based on the following rules:

- [*The TOE shall reject requests for access or services where the presumed source identity 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 identity of the information received by the TOE specifies a broadcast identity;*
- *The TOE shall reject requests for access or services where the presumed source identity of the information received by the TOE specifies a loopback identifier; and*
- *The TOE shall reject requests in which the information received by the TOE contains the route (set of host network identifiers) by which information shall flow from the source subject to the TOE*].

FDP_IFF.1(4)[2] Simple Security attributes (unauthenticated policy)

FDP_IFF.1.1(4) - The TSF shall enforce the [*UNAUTHENTICATED INFORMATION FLOW SFP*] based on the following types of subject and information security attributes:

a) [*Source subject security attributes:*
   - *set of source entity identifiers **(IP address)**; and*
   - [[*configured zone*]].

b) Destination subject security attributes:
   - *Set of destination entity identifiers **(IP address)**; and*
   - [[*configured zone*]].

c) Information security attributes:

[2] The third iteration of FDP_IFF.1 was removed from the ST. Iteration numbering was preserved for document compatibility.

- *presumed identity of source entity1;*
- *identity of destination entity;*
- *transport layer protocol;*
- *source entity service identifier;*
- *destination entity service identifier (e.g., TCP or User Datagram Protocol (UDP) destination port number);*
- [none].
- Stateful packet attributes: [*for IP-based network stacks:*
  - *Connection-oriented protocols:*
    - *sequence number;*
    - *acknowledgement number;*
    - *Flags:*
      - *SYN;*
      - *ACK;*
      - *RST;*
      - *FIN; and*
    - [none].
  - *Connectionless protocols:*
    - *source and destination network identifiers **(IP address)**;*
    - *source and destination service identifiers;*
    - [none];
- for non-IP-based network stacks: [*none*].

FDP_IFF.1.2(4) - **Refinement**: The TSF shall permit an information flow between a **source subject** and a **destination subject** via a controlled operation if the following rules hold:

- [*the presumed identity of the source entity is in the set of source entity identifiers;*
- *the identity of the destination entity is in the set of destination entity 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 [*
  - *Network traffic is received by the TOE on one of its interfaces. These interfaces are grouped into zones,*
  - *The TOE examines the packet security attributes (including, presumed identity of source entity, identity of destination entity, transport layer protocol, source subject service identifier, destination subject service identifier (e.g., TCP or UDP destination port number) and compares the packet to the configured information flow policy rules,*
  - *The TOE performs a stateful inspection of the traffic by examining the packets sequence number, acknowledgement number, and flags for connection-oriented protocols or by examining the source and destination address and the traffic protocol,*

    - *The TOE identifies the configured information flow rule the network traffic meets. The information flow rules identify the actions allowed between configured zones of interfaces,*
- *The TOE passes the network traffic if it meets a configured allow policy and does not meet a configured drop policy]; and*
- *the selected information flow policy rule specifies that the information flow is to be permitted, and what specific operation from FDP_IFC.1 is to be applied to that information flow*].

FDP_IFF.1.3(4) - The TSF shall enforce the [*the following:*

- *fragmentation rule:*
    - *prior to applying the information policy ruleset, the TOE completely reassembles fragmented packets;*
- *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(4), is applied to the packet;*
    - *otherwise, the TSF associates a packet with an allowed established session using the stateful packet attributes*].

FDP_IFF.1.4(4) - The TSF shall provide the following [*the Security Administrator shall have the capability to view all information flows allowed by the information flow policy ruleset before the ruleset is applied*].

FDP_IFF.1.5(4) - The TSF shall explicitly authorize an information flow based on the following rules: [*none*].

FDP_IFF.1.6(4) - The TSF shall explicitly deny an information flow based on the following rules:

- [*The TOE shall reject requests for access or services where the presumed source identity 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 identity of the information received by the TOE specifies a broadcast identity;*
- *The TOE shall reject requests for access or services where the presumed source identity of the information received by the TOE specifies a loopback identifier;*
- *The TOE shall reject requests in which the information received by the TOE contains the route (set of host network identifiers) by which information shall flow from the source subject to the destination subject.*]

FDP_RIP.2 Full residual information protection

FDP_RIP.2.1 - The TSF shall ensure that any previous information content of a resource is made unavailable upon the [deallocation of the resource from] all objects.

5.2.4 Identification and authentication (FIA)

FIA_AFL.1Authentication failure handling

FIA_AFL.1.1 - Refinement: The TSF shall detect when [*a Security administrator configurable positive integer*] **of** unsuccessful authentication attempts occur related to [*administrators attempting to authenticate remotely, and authorized IT entities***].**

FIA_AFL.1.2 – Refinement: When the defined number of unsuccessful authentication attempts has been met ~~or surpassed~~, the TSF shall [*at the option of the Security Administrator prevent the remote administrators or authorized IT entity 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*].

FIA_ATD.1(1) User attribute definition

FIA_ATD.1.1(1) – **Refinement**: The TSF shall maintain the following list of security attributes belonging to an **administrator**:

a) [*user identifier(s):*
- *role;*
- [[none]]; *and*

b) [[*user password*]]].

FIA_ATD.1(2) User Attribute definition

FIA_ATD.1.1(2) **Refinement:** The TSF shall maintain the following list of security attributes belonging to **authorized subjects**:

a) [*subject identity **(IP address/Host Name)**;*
b) [*IKE Security Attributes*].

FIA_UAU.1 Timing of authentication (for TOE services)

FIA_UAU.1.1 - The TSF shall allow [for an IP-based network stack: ICMP, [none]; or for a non-IP based network stack: [*none*]] 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.

FIA_UAU.2 User authentication before any action

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

FIA_UID.2 User identification before any action

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

FIA_USB.1 User-subject binding

FIA_USB.1.1: The TSF shall associate the following user security attributes with subjects acting on the behalf of that user: [

- *Remote IT Entity (VPN Peers): subject identity (IP address/host name), IKE security attributes*
- *User: Username, user password, Role*].

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: [

- *Remote IT Entity (VPN Peers): Whenever the TOE negotiates an IPSec connection with a VPN Peer, the TOE compares the IKE Secure attributes to the internally store peer profile and associates the peer with the profile;*
- *User: Whenever a user presents authentication credentials to the TOE via the TOE CLI, the TOE verifies that the user is a known user. The TOE associates the user with the profile and role within the TOE*].

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: [

- *Remote IT Entity (VPN Peers): the security attributes (Subject identity/ IKE Security Attributes) can only be changed by renegotiating a IPSec connection with the remote IT entity;*
- *User: the security attributes associated with a user may only be changed by a Security Administrator*].

5.2.5 Security management (FMT)

FMT_MOF.1(1) Management of security functions behavior (TSF non-Cryptographic Self-test)

FMT_MOF.1.1(1) - The TSF shall restrict the ability to determine and modify the behavior of the functions:

- *[TSF Self-Test (FPT_TST_(EXT).1)*]

to [*the Security Administrator*].

FMT_MOF.1(2) Management of security functions behavior (Cryptographic Self-test)

FMT_MOF.1.1(2) - The TSF shall restrict the ability to enable~~, disable~~ the functions

- [*Crypto Self-Test (FPT_TST.1(1), and Key Generation Self-Test (FPT_TST.1(2)*]

to [*the Cryptographic Administrator*].

FMT_MOF.1(3) Management of security functions behavior (audit and alarms)

FMT_MOF.1.1(3) - The TSF shall restrict the ability to enable, disable, determine and modify the behavior of the functions

- [*Security Audit (FAU_SAR)**.*]** to [*an Administrator*].

FMT_MOF.1(4) Management of security functions behavior (audit and alarms)

FMT_MOF.1.1(4) - The TSF shall restrict the ability to enable, disable, determine and modify the behavior of the functions

- [*Security Audit Analysis (FAU_SAA); and*
- *Security Audit (FAU_SEL)*]

to [*the Security Administrator*].

FMT_MOF.1(5) Management of security functions behavior (audit and alarms)

FMT_MOF.1.1(5) - The TSF shall restrict the ability to enable, or disable the functions

- [*Security Alarms (FAU_ARP)*]

to [*the Security Administrator*].

FMT_MOF.1(6) Management of security functions behavior (available TOE-services for unauthenticated users)

FMT_MOF.1.1(6) - The TSF shall restrict the ability to enable, disable the functions

- [[*for an IP-based network stack: ICMP*, [none], or for a non-IP-based network stack: [*none*]].

to [*the Security Administrator*].

FMT_MOF.1(7) Management of security functions behavior (quota mechanism)

FMT_MOF.1.1(7) - The TSF shall restrict the ability to determine the behavior *of* the functions

- [*Controlled connection-oriented resource allocation (FRU_RSA.1(2));*
- *An administrator-specified network identifier;*
- *set of administrator-specified network identifiers;*
- *administrator-specified period of time*]

to [*the Security Administrator*].

FMT_MOF.1(8) Management of security functions behavior (Authentication Attempts)

FMT_MOF.1.1(8) - The TSF shall restrict the ability to enable, disable, determine and modify the behavior of the functions

- [*Authentication failure handling (FIA_AFL.1.2) configurable integer of unsuccessful authentication attempts that occurs related to a user's authentication* to [*the Security Administrator*]].

FMT_MSA.1(1) Management of security attributes

FMT_MSA.1.1(1) - The TSF shall enforce the [*UNAUTHENTICATED INFORMATION FLOW SFP, UNAUTHENTICATED TOE SERVICES SFP*] to restrict the ability to [*manipulate*] the security attributes [*referenced in the indicated polices*] to [*the Security Administrator*].

FMT_MSA.1(2) Management of security attributes

FMT_MSA.1.1(2) - The TSF shall enforce the [*VPN SFP*] to restrict the ability to [*manipulate*] the security attributes [*referenced in the indicated polices*] to [~~*an*~~ ***the security*** *Administrator*].

FMT_MSA.1(3) Management of security attributes

FMT_MSA.1.1(3) - The TSF shall enforce the [*UNAUTHENTICATED INFORMATION FLOW SFP*] to restrict the ability to [query, modify, delete, [*no other operations*]] the security attributes [*settable percentage of storage capacity* [*the security attributes contained within administratively configured policy rule-sets*]] to [*Security Administrator* [*no other roles*]].

FMT_MSA.3(1) Static attribute initialization

FMT_MSA.3.1(1) - **Refinement:** The TSF shall enforce the [*VPN SFP*] to provide *restrictive* default values for the (**security attributes) information flow policy ruleset** that **is** (**are**) used to enforce the SFP.

FMT_MSA.3.2(1) 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_MSA.3(2) Static attribute initialization

FMT_MSA.3.1(2) – **Refinement:** The TSF shall enforce the [*UNAUTHENTICATED INFORMATION FLOW SFP*] to provide *restrictive* default values for the **information flow policy ruleset** that is used to enforce the SFP.

FMT_MSA.3.2(2) - 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_MSA.3(3) Static attribute initialization (services)

FMT_MSA.3.1(3) – **Refinement:** The TSF shall enforce the [*UNAUTHENTICATED TOE SERVICES SFP*] to provide *restrictive* default values for (**security attributes) the set of TOE services available to unauthenticated users (that are used to enforce the SFP).**

FMT_MSA.3.2(3) - 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_MTD.1(1) Management of TSF data

FMT_MTD.1(1) - **Refinement:** The TSF shall restrict the ability to [query, modify, delete, clear, [*no other operations*] **all** the [*TSF data except cryptographic security data, the time and date used to form the time stamps in FPT_STM.1, **and the deletion of audit data***] to [*the Security administrator*].
Application Note: Note that FMT_MTD.1(3) grants the time and date privileges to the Security Administrator. This does not conflict with this requirement as the original wording was intended to specify privileges of all administrators, and it was made more restrictive for this Security Target.

FMT_MTD.1(2) Management of TSF data

FMT_MTD.1.1(2) - The TSF shall restrict the ability to *modify* the [*cryptographic security data*] to [*the Cryptographic Administrator*].

FMT_MTD.1(3) Management of TSF data

FMT_MTD.1.1(3) - The TSF shall restrict the ability to [set] the [*time and date used to form the time stamps in FPT_STM.1*] to [*the Security Administrator ~~or authorized IT entity~~*].

FMT_MTD.1(4) Management of TSF data

FMT_MTD.1.1(4) – The TSF shall restrict the ability to [query, modify, delete, create, [*none*]] the [*information flow policy rules*] to [*the Security Administrator*].

FMT_MTD.2(1) Management of limits on TSF data

FMT_MTD.2.1(1) - The TSF shall restrict the specification of the limits for [*quotas on transport-layer connections*] to [*the Security Administrator*].

FMT_MTD.2.2(1) - The TSF shall take the following actions, if the TSF data are at, or exceed, the indicated limits: [

- *For maximum sessions, new connections are not accepted /created;*
- *For maximum incomplete sessions, the oldest half-opened sessions are deleted. If there are no half-open sessions, new connections are not accepted/created;*

- *For maximum high/low sessions over one minute, the oldest half-opened sessions are deleted. If there are no half-open sessions, new connections are not accepted/created*].

FMT_MTD.2(2) Management of limits on TSF data

FMT_MTD.2.1(2) - The TSF shall restrict the specification of the limits for [*quotas on controlled connection-oriented resources*] to [*the Security Administrator*].

FMT_MTD.2.2(2) - The TSF shall take the following actions, if the TSF data are at, or exceed, the indicated limits: [

- *For maximum sessions, new connections are not accepted /created;*
- *For maximum incomplete sessions, the oldest half-opened sessions are deleted. If there are no half-open sessions, new connections are not accepted/created;*
- *For maximum high/low sessions over one minute, the oldest half-opened sessions are deleted. If there are no half-open sessions, new connections are not accepted/created*].

FMT_MTD.2(3) Management of limits on TSF data

FMT_MTD.2.1(3) - The TSF shall restrict the specification of the limits for [*percentage of storage capacity for audit records*] to [*the Security Administrator*].

FMT_MTD.2.2(3) - The TSF shall take the following actions, if the TSF data are at, or exceed, the indicated limits: [*Terminates all active processes or overwrites the oldest audit records with the newest records, per administrative configuration*].

FMT_REV.1 Revocation

FMT_REV.1.1 –The TSF shall restrict the ability to revoke security attributes associated with the [*users, information flow policy ruleset, services available to unauthenticated users***,** [subjects, objects] within the TSC to [*the Security Administrator*[3]].

FMT_REV.1.2 - **Refinement:** The TSF shall **immediately** enforce the:

- [*revocation of a user's role (Security Administrator, Cryptographic Administrator, Audit Administrator);*
- *changes to the information flow policy ruleset when applied;*
- *disabling of a service available to unauthenticated users;*
- *changes to the set of security associations with peer TOEs and*
- [none]].

FMT_SMF.1 Specification of Management Functions

[3] No role can revoke users's attributes (role or password). This must be done by the SuperAdmin outside of the evaluated configuration.

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

1. *restrict the ability to ~~invoke~~ determine and modify the behavior of the functions TSF Self-Test (FPT_TST_(EXT).1) to the Security Administrator;*
2. *restrict the ability to enable, ~~disable~~ the functions TSF Self-Test (FPT_TST.1(1) and FPT_TST.1(2)) to the Cryptographic Administrator;*
3. *restrict the ability to enable, disable, determine and modify the behavior of the functions Security Audit (FAU_SAR.1, FAU_SAR.2, FAU_SAR.3) to an ~~Audit~~ Administrator;*
4. *restrict the ability to enable, disable, determine and modify the behavior of the functions Security Audit Analysis (FAU_SAA); and Security Audit (FAU_SEL) to the Security Administrator;*
5. *restrict the ability to enable, or disable the functions Security Alarms (FAU_ARP) to the Security Administrator*
6. *restrict the ability to determine the behavior of the functions: Controlled connection-oriented resource allocation (FRU_RSA.1(2)); an administrator-specified network identifier; set of administrator-specified network identifiers; administrator-specified period of time to the Security Administrator.*
7. *enforce administrator-specified maximum quotas of the following resources: [controlled connection-oriented resources] that users associated with [an administrator-specified network identifier and a set of administrator-specified network identifiers] can use over an administrator-specified period of time.*
8. *enforce the [UNAUTHENTICATED INFORMATION FLOW SFP] to provide restrictive default values security attributes that are used to enforce the SFP;*
9. [*none*]
10. *restrict the ability to* [query, modify, delete, clear, [*none*]] *all the [TSF data except cryptographic security data and the time and date used to form the time stamps in FPT_STM.1] to the **Security** administrator~~s~~ ~~or authorized IT entities (FMT_MTD.1(1))~~ .*
11. *restrict the ability to modify the cryptographic security data to the Cryptographic Administrator (FMT_MTD.1(2));*
12. *restrict the ability to set the time and date used to form the time stamps in FPT_STM.1 to the Security Administrator ~~or authorized IT entity~~.*
13. *[none]*
14. *[none]*
15. *restrict the ability to query, modify, delete, create,* [none] *the [information flow policy rules] to [the Security Administrator] (FMT_MTD.1.1(4)).*
16. *restrict the ability to revoke security attributes associated with the* [users, subjects, objects, [*other additional resources*]] *within the TSC to Security administrator (FMT_REV.1)*
17. *restrict the specification of the limits for quotas on transport-layer connections to the Security Administrator (FMT_MTD.2 (1));*
18. *restrict the specification of the limits for quotas on controlled connection-oriented resources to the Security Administrator (FMT_MTD.2 (2));*
19. *restrict the specification of the limits for percentage of storage capacity for audit records (FMT_MTD.2(3)) to the Security Administrator.*

20. *restrict the ability to enable, disable, determine and modify the behavior of the functions of Authentication failure handling (FIA_AFL.1.2) to configure an integer of unsuccessful authentication attempts that occurs related to a user's authentication to the Security Administrator.*
21. [*none*]].
22. *restrict the ability to enable, disable the* functions [[for an IP-based network stack: ICMP, [[*none*], or *for a non-IP-based network stack*:[*none*]] *to [the Security Administrator]*.
23. ~~*restrict the ability to determine the behavior of the functions by an administrator-specified network identifier; set of administrator specified network identifiers; administrator specified period of time] to [the Security Administrator];*~~
24. *enforce the [VPN SFP] to restrict the ability to manipulate the security attributes referenced in the indicated polices to the Security Administrator;*
25. *enforce the [VPN SFP] to provide restrictive default values for the information flow policy rule set security attributes that is used to enforce the SFP;*
26. *enforce the [UNAUTHENTICATED TOE SERVICES SFP] to provide restrictive default values security attributes that are used to enforce the SFP;*
27. *restrict the ability to query, modify, delete, create, [none] the VPN Policy rules to the Security Administrator*;
28. *[No additional Security Functions]*.

FMT_SMR.2 Restrictions on security roles

FMT_SMR.2.1 - The TSF shall maintain the roles:

- [*Security Administrator;*
- *Cryptographic Administrator (i.e., users authorized to perform cryptographic initialization and management functions);*
- *Audit Administrator;*
- *Authorized IT entities; and*
- [none]].

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

- [*All roles shall be able to administer the TOE locally;*
- *all roles shall be able to administer the TOE remotely;*
- *all roles are distinct; that is, there shall be no overlap of operations performed by each role, with the following exceptions:*
- *all administrators can review the audit trail; and*
- *all administrators can invoke the self-tests*] are satisfied.

5.2.6 Protection of the TSF (FPT)

FPT_FLS.1 Failure with preservation of secure state

FPT_FLS.1.1 The TSF shall preserve a secure state when the following types of failures occur: [*all failures*].

FPT_ITA.1 Inter-TSF availability within a defined availability metric

FPT_ITA.1.1 The TSF shall ensure the availability of [*network control data associated with the TOE supported routing protocols*] provided to a remote trusted IT product within [*the availability metrics defined in the following routing protocols:*

- *BGP: None of the defined timers (ConnectRetry, Hold Time, KeepAlive, MinASOriginationInterval, and MinRouteAdvertisementInterval) are exceeded;*
- *PIM: None of the defined timers (holdtime timer, hello timer, expiration timer, and register-suppression timer) are exceeded;*
- *IS-IS: None of the defined timers (recallTimer, idleTimer, initialMinimumTimer, reserveTimer, holdingTimer) are exceeded;*
- *OSPF: None of the defined timers (Hello Timer, Wait Timer, Inactivity Timer, interval timer, retransmission timer) are exceeded;*
- *RIP: None of the defined timers (timeout timer, garbage-collection timer, routing-update timer) are exceeded;*
- *MPLS: None of the defined TTLs are exceeded*]

given the following conditions [*The TOE is persistently connected to a network*].

FPT_ITC.1 Inter-TSF confidentiality during transmission

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

FPT_ITI.1 Inter-TSF detection of modification

FPT_ITI.1.1 The TSF shall provide the capability to detect modification of all TSF data during transmission between the TSF and a remote trusted IT product within the following metric: [*immediately when TSF data is received by the TOE*].

FPT_ITI.1.2 The TSF shall provide the capability to verify the integrity of all TSF data transmitted between the TSF and a remote trusted IT product and perform [*disregard the information*] if modifications are detected.

FPT_RCV.1 Manual Recovery

FPT_RCV.1.1 **Refinement:** After **a** [failure or service discontinuity], the TSF shall enter a maintenance mode where the ability to return **the TOE** to a secure state is provided.

FPT_RCV.2 Automated Recovery

FPT_RCV.2.1 When automated recovery from [*a failure or service discontinuity*] is not possible, the TSF shall enter a maintenance mode where the ability to return to a secure state is provided.

FPT_RCV.2.2 For [

- *Dual RP/ESP TOE configuration (ASR 1006)*
  - *Any single hardware failure on the active RP*
  - *Any single hardware failure on the standby RP*
  - *Any single hardware failure on the active ESP*
  - *Any single hardware failure on the standby ESP*
  - *Any single software failure on the active RP*
  - *Any single software failure on the standby RP*
  - *Any single software failure on the active ESP*
  - *Any single software failure on the standby ESP*
- *Any TOE configuration (ASR1002, ASR 1002f, ASR 1004)*
  - *The TOE experiences a RNG failure in which the consecutive identical random numbers are generated,]*

the TSF shall ensure the return of the TOE to a secure state using automated procedures.

FPT_RPL.1 Replay detection

FPT_RPL.1.1 - The TSF shall detect replay for the following entities: [*TSF data and security attributes*].
FPT_RPL.1.2 - The TSF shall perform: [reject data, audit event and [*no other actions*]] when replay is detected.

FPT_STM.1 Reliable time stamps

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

FPT_TDC.1 Inter-TSF basic TSF data consistency

FPT_TDC.1.1 The TSF shall provide the capability to consistently interpret [*routing data associated with the TOE supported protocols listed in FPT_PRO_(EXT).1*] when shared between the TSF and another trusted IT product.

FPT_TDC.1.2 The TSF shall use [*the routing data is interpreted according to the rules of the protocol with which it is associated (supported protocols are listed in FPT_PRO_(EXT).1).*] when interpreting the TSF data from another trusted IT product.

FPT_TST.1(1) TSF Testing (for cryptography)

FPT_TST.1.1(1) **Refinement:** The TSF shall run a suite of self tests **in accordance with FIPS PUB 140-2 and Appendix A of this profile** during initial start-up **(on power on)**, at the request of the **cryptographic administrator (on demand)**, **under various conditions**

defined in section 4.9.1 of FIPS 140-2, and periodically (at least once a day) to demonstrate the correct operation of the **following cryptographic functions:**

a) **key error detection;**
b) **cryptographic algorithms;**
c) **RNG/PRNG**

FPT_TST.1.2(1) **Refinement:** The TSF shall provide authorized **cryptographic administrators** with the capability to verify the integrity of **TSF data related to the cryptography by using TSF-provided cryptographic functions**.

FPT_TST.1.3(1) **Refinement:** The TSF shall provide authorized **cryptographic administrators** with the capability to verify the integrity of stored TSF executable code **related to the cryptography by using TSF-provided cryptographic functions**.

FPT_TST.1(2) TSF Testing (for key generation components)

FPT_TST.1.1(2) **Refinement**: The TSF shall **perform** self tests **immediately after generation of a key** to demonstrate the correct operation **of each key generation component**. **If any of these tests fails, that generated key shall not be used, the cryptographic module shall react as required by FIPS PUB 140-2 for failing a self-test, and this event will be audited**

FPT_TST.1.2(2) **Refinement:** The TSF shall provide authorized **cryptographic administrators** with the capability to verify the integrity of TSF data **related to the key generation by using TSF-provided cryptographic functions**.

FPT_TST.1.3(2) **Refinement:** The TSF shall provide authorized **cryptographic administrators** with the capability to verify the integrity of stored TSF executable code **related to the key generation by using TSF-provided cryptographic functions**.

5.2.7 Resource Utilization (FRU)

FRU_RSA.1(1) Maximum quotas

FRU_RSA.1.1(1) - **Refinement:** The TSF shall enforce maximum quotas of the following resources: [transport-layer representation] that **a *source subject identifier/users*** can use *over a specified period of time*.

FRU_RSA.1(2) Maximum quotas (controlled connection-oriented quotas)

FRU_RSA.1.1(2) – **Refinement:** The TSF shall enforce **administrator-specified** maximum quotas of the following resources: [*controlled connection-oriented resources*] that ***users associated with an administrator-specified network identifier and a set of administrator-specified network identifiers*** can use *over **an administrator**-specified period of time*.

5.2.8 TOE Access (FTA)

FTA_SSL.1 TSF-initiated session locking

FTA_SSL.1.1 – **Refinement:** The TSF shall lock **a local** interactive session after [*a Security Administrator-specified time period of inactivity*] by:

a) clearing or overwriting display devices, making the current contents unreadable;
b) disabling any activity of the user's data access/display devices other than unlocking the session.

FTA_SSL.1.2 - The TSF shall require the user to re-authenticate prior to unlocking the session.

FTA_SSL.2 User-initiated locking

FTA_SSL.2.1 – **Refinement:** The TSF shall allow user-initiated locking of the user's own **local** interactive session, by:

a) clearing or overwriting display devices, making the current contents unreadable;
b) disabling any activity of the user's data access/display devices other than unlocking the session.

FTA_SSL.2.2 - The TSF shall require the user to re-authenticate prior to unlocking the session.

FTA_SSL.3 TSF-initiated termination

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

FTA_TAB.1 Default TOE access banners

FTA_TAB.1.1 - **Refinement:** Before establishing **a user/administrator** session the TSF shall display **only a Security Administrator-specified** advisory **notice and consent** warning message regarding unauthorized use of the TOE.

FTA_TSE.1 TOE session establishment

FTA_TSE.1.1 - **Refinement:** The TSF shall be able to deny establishment **of an administrator session** based on [*location, time, and day*].

5.2.9 Trusted Path/Channels (FTP)

FTP_ITC.1(1) Inter-TSF trusted channel (Prevention of Disclosure)

FTP_ITC.1.1(1) - **Refinement:** The TSF shall **use encryption to** provide a **trusted** communication channel between itself and **authorized IT entities** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from disclosure.

FTP_ITC.1.2(1) **Refinement:** The TSF shall permit *the TSF*, ***or the authorized IT entities*** to initiate communication via the trusted channel.

FTP_ITC.1.3(1) - The TSF shall initiate communication via the trusted channel for [all authentication functions, [[*exchange of network control information (PIM, IS-IS, BGP, OSPF, RIP, and MPLS)*]].

FTP_ITC.1(2) Inter-TSF trusted channel (Detection of Modification)

FTP_ITC.1.1(2) - **Refinement:** The TSF shall **use a cryptographic signature** to provide a **trusted** communication channel between itself and **authorized IT entities** that is logically distinct from other communication channels and provides assured identification of its end points and **detection of the modification of data**.

FTP_ITC.1.2(2) - **Refinement:** The TSF shall permit *the TSF*, ***or the authorized IT entities*** to initiate communication via the trusted channel.

FTP_ITC.1.3(2) - The TSF shall initiate communication via the trusted channel for [all authentication functions, [[*exchange of network control information (PIM, IS-IS, BGP, OSPF, RIP, and MPLS)*]].

FTP_TRP.1(1) Trusted path

FTP_TRP.1.1(1) - **Refinement:** The TSF shall provide a **an encrypted** communication path between itself and *remote* **administrators, authenticated** users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from disclosure.

FTP_TRP.1.2(1) - The TSF shall permit *remote users* to initiate communication via the trusted path.

FTP_TRP.1.3(1) – **Refinement:** The TSF shall require the use of the trusted path for *user authentication*, all remote administration actions, [none].

FTP_TRP.1(2) Trusted path

FTP_TRP.1.1(2) - **Refinement:** The TSF shall **use a cryptographic signature to** provide a communication path between itself and remote **Administrators and authenticated users** that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from **detection of the modification of data**.

FTP_TRP.1.2(2) - The TSF shall permit *remote users* to initiate communication via the trusted path.

FTP_TRP.1.3(2) – **Refinement:** The TSF shall require the use of the trusted path for *initial user authentication*, all remote administration actions, and [none].

5.2.10 Extended Components Definition

This Security Target contains fourteen Security Functional Requirements that are not drawn from existing CC part 2 Security Function Requirements.

The identification structure of each Security Functional Requirement is modeled after the Security Functional Requirements included in CC part 2. The identification structure includes the following:

A. Class – The extended SFRs included in this ST are part of the FAU, FCS, FIA, or FPT class of requirements

B. Family – The extended SFRs included in this ST are part of several SFR families

C. Component – The extended SFRs are at several component levels, including, 1, 2, 3, 4, and 5.

FAU_ARP_ACK_(EXT).1 Security alarm acknowledgement

FAU_ARP_ACK_(EXT).1.1 – The TSF shall display the alarm message identifying the potential security violation and make accessible the audit record contents associated with the auditable event(s) until it has been acknowledged. **If the Security Administrator configures the TOE to generate an optional audible alarm,** an audible alarm will sound until acknowledged **by an administrator. Once the alarm is acknowledged, it will be reset to zero.**

FAU_ARP_ACK_(EXT).1.2 – The TSF shall display an acknowledgement message 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:

a) local console, and
b) remote administrator sessions that received the alarm.

FAU_STG_(EXT).4 Site-Configurable Prevention of Audit Loss

FAU_STG_(EXT).4.1 - **Refinement**: The TSF shall provide the **Security** Administrator the capability to select one or more of the following actions:

a) prevent auditable events, except those taken by the Security Administrator and Audit Administrator,
b) *overwrite the oldest stored audit records* and
c) [no other actions]

to be taken if the audit trail is full.

FAU_STG_(EXT).4.2 - **Refinement**: The TSF shall **enforce the Security Administrator's selection(s)** if the audit trail is full.

FCS_BCM_(EXT).1 Explicit: Baseline Cryptographic Module

FCS_BCM_(EXT).1.1 - All FIPS-approved cryptographic functions implemented by the TOE shall be implemented in a cryptomodule that is FIPS 140-2 validated, and perform the specified cryptographic functions in a FIPS-approved mode of operation. The FIPS 140-2 validation shall include an algorithm validation certificate for all FIPS-approved cryptographic functions implemented by the TOE.

FCS_BCM_(EXT).1.2 All cryptographic modules implemented in the TOE [As a combination of hardware and software shall have a minimum overall rating of FIPS PUB 140-2, Level 1 and also meet FIPS PUB 140-2, Level 3 for the following: Cryptographic Module Ports and Interfaces; Roles, Services and Authentication; ~~***Cryptographic Key Management;***~~ [4]and Design Assurance.]

FCS_CKM_(EXT).2 Explicit: Cryptographic Key Handling and Storage

FCS_CKM_(EXT).2.1 - The TSF shall perform a key error detection check on each transfer of key (internal, intermediate transfers).

.

FCS_CKM_(EXT).2.2 - The TSF shall store persistent secret and private keys when not in use in encrypted form or using split knowledge procedures.

FCS_CKM_(EXT)_2.3 - The TSF shall destroy non-persistent cryptographic keys after a cryptographic administrator-defined period of time of inactivity.

FCS_CKM_(EXT).2.4 The TSF shall prevent archiving of expired (private) signature keys.

FCS_COP_(EXT).1 Explicit: Random Number Generation

FCS_COP_(EXT).1.1 - **The TSF shall perform all random number generation (RNG) services in accordance with a FIPS-approved RNG** [*FIPS approved 3-key TDES based ANSI X9.31 compliant pseudo RNG*] **seeded by** [a combination of hardware-based and software-based entropy sources.]

FCS_COP_(EXT).1.2 **-** The TSF shall defend against tampering of the random number generation (RNG)/ pseudorandom number generation (PRNG) sources.

FCS_IKE_(EXT).1 Internet Key Exchange

FCS_IKE_(EXT).1.1 –The TSF shall provide cryptographic key establishment techniques in accordance with RFC 2409 as follows(s):

- Phase 1, the establishment of a secure authenticated channel between the TOE and another remote VPN endpoint, shall be performed using one of the following, as configured by the security administrator:

[4] Note that there are no testable requirements to differentiate Level 1 through 4 for our implementation of cryptographic keys and key management, so for Section 7, we can't say AS07.30 applies/passes and then claim Level 3 for that section.

  - Main Mode
  - Aggressive Mode

- Phase 2, negotiation of security services for IPsec, shall be done using Quick Mode, using SHA-1 as the pseudo-random function. Quick Mode shall generate key material that provides perfect forward secrecy. The use of SHA-256 and SHA-384 as the PRF in IKEv1 KDF is also allowed.

FCS_IKE_(EXT).1.2 – The TSF shall require the x of g^xy be randomly generated using a FIPS-approved random number generator when computation is being performed. The minimum size of x shall be twice the number of bits of the strength level associated with the negotiated DH group per table 2 of NIST SP 800-57. The nonce sizes are to be between 8 and 256 bytes. Nonces shall be generated in a manner such that the probability that a specific nonce value will be repeated during the life a specific IPsec SA is less than 1 in 2^(bit strength of the negotiated DH group)**.**

FCS_IKE_(EXT).1.3 - When performing authentication using pre-shared keys, the key shall be generated using the FIPS approved random number generator specified in FCS_COP_(EXT).1.1.

FCS_IKE_(EXT).1.4 - The TSF shall compute the value of SKEYID (as defined in RFC 2409), using SHA-1 as the pseudo-random function. The use of SHA-256 and SHA-384 as the PRF in IKEv1 KDF is also allowed. The TSF shall be capable of authentication using the methods for

- Signatures: SKEYID = prf(Ni_b | Nr_b, g^xy)
- Pre-shared keys: `SKEYID = prf(pre-shared-key, Ni_b | Nr_b)`
- [Authentication using Public key encryption, computing SKEYID as follows: SKEYID = prf(prf(Ni_b | Nr_b), CKY-I | CKY-R), no other authentication methods]

FCS_IKE_(EXT).1.5 - The TSF shall compute authenticated keying material as follows:

- SKEYID_d = prf(SKEYID, g^xy | CKY-I | CKY-R | 0)
- SKEYID_a = prf(SKEYID, SKEYID_d | g^xy | CKY-I | CKY-R | 1)
- SKEYID_e = prf(SKEYID, SKEYID_a | g^xy | CKY-I | CKY-R | 2)
- [none]]

FCS_IKE_(EXT).1.6 - To authenticate the Phase 1 exchange, the TSF shall generate HASH_I if it

is the initiator, or HASH_R if it is the responder as follows:

- HASH_I = prf(SKEYID, g^xi | g^xr | CKY-I | CKY-R | SAi_b | IDii_b)
- HASH_R = prf(SKEYID, g^xr | g^xi | CKY-R | CKY-I | SAi_b | IDir_b)

FCS_IKE_(EXT).1.7 - The TSF shall be capable of authenticating IKE Phase 1 using the following methods as defined in RFC 2409, as configured by the security administrator:

a) **Authentication with digital signatures**: The TSF shall use [RSA, "no other digital signature algorithms"]
b) when an RSA signature is applied to HASH I or HASH R it must be first PKCS#1 encoded. The TSF shall check the HASH_I and HASH_R values sent against a computed value to detect any changes made to the proposed transform negotiated in phase one. If changes are detected the session shall be terminated and an alarm shall be generated.
c) [[*X.509 certificates Version 3* [no other versions]] X.509 V3 implementations, if implemented, shall be capable of checking for validity of the certificate path, and at option of SA, check for certificate revocation.
d) **Authentication with a pre-shared key**: The TSF shall allow authentication using a pre-shared key.

FCS_IKE_(EXT).1.8. - The TSF shall compute the hash values for Quick Mode in the following way

HASH(1) = prf(SKEYID_a, M-ID |[any ISAKMP payload after
HASH(1) header contained in the message)]
HASH(2) = prf(SKEYID_a, M-ID | Ni_b | [any ISAKMP payload after HASH(2) header contained in the message)]
HASH(3) = prf(SKEYID_a, 0 | M-ID | Ni_b | Nr_b)

FCS_IKE_(EXT).1.9 - The TSF shall compute new keying material during Quick Mode as follows:

[when using perfect forward secrecy
KEYMAT = prf(SKEYID_d, g(qm)^xy | protocol | SPI | Ni_b | Nr_b),
When perfect forward secrecy is not used
KEYMAT = prf(SKEYID_d | protocol | SPI | Ni_b | Nr_b)]

FCS_IKE_(EXT).1.10 - The TSF shall at a minimum, support the following ID types: *ID_IPV4_ADDR, ID_IPV6_ADDR, ID_FQDN, ID_USER_FQDN,* [ID_IPV4_ADDR_SUBNET, ID_IPV6_ADDR_SUBNET, ID_IPV4_ADDR_RANGE, ID_IPV6_ADDR_RANGE, ID_DER_ASN1_DN, ID_DER_ASN1_GN, ID_KEY_ID].

FCS_GDOI_(EXT).1 Group Domain of Interpretation

FCS_GDOI_(EXT).1.1 - The TSF shall provide negotiation of security services for IPsec in accordance with RFC 3457 as an extension of phase 2 of the protocol defined in RFC 2409, negotiation of security services for IPsec.

FCS_GDOI_(EXT).1.2 – The TSF shall provide the "GROUPKEY-PULL" registration protocol as defined in RFC 3457 that protects the key agreement packets providing confidentiality and integrity for the communications between a new group member and the group controller.

FCS_GDOI_(EXT).1.3 – The TSF shall provide the "GROUPKEY-PUSH" rekey protocol as defined in RFC 3457 that protects the key agreement packets as they pass from the controller

to the members, for confidentiality using the AES encryption algorithm specified in FCS_COP.1.1(1).

Extended Requirements Rationale – FCS_GDOI_(EXT).1:

A. Class – The FCS class of SFRs identifies cryptographic functionality provided by the TOE. FCS_GDOI_(EXP).1 describes the cryptographic functionality associated with the Group Domain of Interpretation extension of IPSec (defined in RFC 3457) provided by the TOE. This is cryptographic functionality and consistent with the FCS class of SFRs.

B. Family – This is a newly created SFR family, GDOI. This family was created to describe the Group Domain of Interpretation functionality provided by the TOE. There is not a family defined in the Common Criteria Part 2 to address Group Domain of Interpretation. This is why the new family was created.

C. Component – This is the only component in the family. This is why the component is identified as "1."

Management – FCS_GDOI_(EXP).1:

There are no management activities foreseen.

FIA_UAU_(EXT).5 Multiple authentication mechanisms

FIA_UAU_(EXT).5.1 - The TSF shall provide a local authentication mechanism, [none] to perform user authentication.

FPT_HA_(EXT).1 High Availability

FPT_HA_(EXT).1.1 - The TSF provides hardware failover for any single hardware or software fault within the TSF for any TOE configuration which includes dual RPs/ESPs.

Extended Requirements Rationale – FPT_HA_(EXT).1:

A. Class – The FPT class of SFRs identifies TSF integrity functionality provided by the TOE. FPT_HA_(EXP).1 describes the high availability functionality provided by the TOE. High availability ensures that the integrity of the TOE and security functionality provided by the TOE are maintained even during specific failure events. This is integrity functionality and consistent with the FPT class of SFRs.

B. Family – This is a newly created SFR family, HA. This family was created to describe the high availability functionality provided by the TOE. There is not a family defined in the Common Criteria Part 2 to address the high availability functionality provided by the TOE. This is why the new family was created.

C. Component – This is the only component in the family. This is why the component is identified as "1."

Management – FPT_HA_(EXP).1

There are no management activities foreseen.

FPT_PRO_(EXT).1 Standard Protocol Usage

FPT_PRO_(EXT).1 - The TSF shall utilize the standard protocol mechanisms within the standard protocols [

a) *RFC 1771: BGP;*
b) *RFC 4601: PIM;*
c) *RFC 1142: IS-IS;*
d) *RFC 2328: OSPF;*
e) *RFC 2453: RIP;*
f) *RFC 3031: MPLS*].

FPT_TST_(EXT).1 Extended: TSF Testing

FPT_TST_(EXT).1.1 The TSF shall run a suite of self tests during the initial start-up and also either periodically during normal operation, or at the request of an authorized administrator to demonstrate the correct operation of the TSF.

FPT_TST_(EXT).1.2 The TSF shall provide authorized administrators with the capability to verify the integrity of stored TSF executable code through the use of the TSF-provided cryptographic services.

5.3 IT Environment Security Functional Requirements

This section identifies the Security Functional Requirements for the IT Environment. The IT Environment Security Functional Requirements that appear in the table, "Security Functional Requirements" are described in more detail in the following subsections.

Table 23 Security Functional Requirements

Functional Component	
SFR Component ID	**Component Name**
Security Functional Requirements Directly Drawn from CC Part 2	
FTA_SSL.1	TSF-initiated session locking
FTP_ITC.1(1)	Inter-TSF trusted channel
FTP_ITC.1(2)	Inter-TSF trusted channel
FTP_TRP.1(1)	Trusted path
FTP_TRP.1(2)	Trusted path

FTA_SSL.1 TSF-initiated session locking

FTA_SSL.1.1 – **Refinement:** The **IT Environment** shall lock **a local** interactive session after [*a Security Administrator-specified time period of inactivity*] by:

a) clearing or overwriting display devices, making the current contents unreadable;

b) disabling any activity of the user's data access/display devices other than unlocking the session.

FTA_SSL.1.2 - The TSF shall require the user to re-authenticate prior to unlocking the session.

FTP_ITC.1(1) Inter-TSF trusted channel (Prevention of Disclosure)

FTP_ITC.1.1(1) - **Refinement:** The **IT Environment** shall provide a **trusted** communication channel between itself and the **TSF** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from disclosure.

FTP_ITC.1.2(1) - **Refinement:** The **IT Environment** shall permit *the TSF* **or the IT Environment** to initiate communication via the trusted channel.

FTP_ITC.1.3(1) - The **IT Environment** shall initiate communication via the trusted channel for [all authentication functions, [[*exchange of routing protocol information*].

FTP_ITC.1(2) Inter-TSF trusted channel (Detection of Modification)

FTP_ITC.1.1(2) - **Refinement:** The **IT Environment** shall provide **an encrypted** communication channel between itself and **the TSF** that is logically distinct from other communication channels and provides assured identification of its end points and **detection of the modification of data.**

FTP_ITC.1.2(2) - **Refinement:** The **IT Environment** shall permit *the TSF*, or **the IT Environment** to initiate communication via the trusted channel.

FTP_ITC.1.3(2) - The **IT Environment** shall initiate communication via the trusted channel for [all authentication functions, [[*exchange of routing protocol information*]].

FTP_TRP.1(1) Trusted path (Prevention of Disclosure)

FTP_TRP.1.1(1) - **Refinement:** The **IT Environment** shall provide an **encrypted** communication path between itself and **the TSF** that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from modification or disclosure.

FTP_TRP.1.2(1) - The **IT Environment** shall permit *remote administrators* **of the TSF** to initiate communication to the TSF via the trusted path.

FTP_TRP.1.3(1) – **Refinement:** The **IT Environment** shall **initiate** the use of the trusted path for **all remote administration actions**, [*administrative user authentication*].

FTP_TRP.1(2) Trusted path (Detection of Modification)

FTP_TRP.1.1(2) - **Refinement:** The **IT Environment** shall provide an **encrypted** communication path between itself and **the TSF** that is logically distinct from other communication paths and provides assured identification of its end points **and detection of the modification of data.**

FTP_TRP.1.2(2) - **Refinement:** The **IT Environment** shall permit *remote* **administrators of the TSF** to initiate communication **to the TSF** via the trusted path.

FTP_TRP.1.3(2) – **Refinement:** The **IT Environment** shall **initiate** the use of the trusted path for *user authentication, all remote administration actions*, [none].

5.4 TOE SFR Hierarchies and Dependencies

This section of the Security Target demonstrates that the identified TOE and IT Security Functional Requirements include the appropriate hierarchical SFRs and dependent SFRs. The following table lists the TOE Security Functional Components and the Security Functional Components each are hierarchical to and dependent upon and any necessary rationale.

N/A in the Rationale column means the Security Functional Requirement has no dependencies and therefore, no dependency rationale is required. Satisfied in the Rationale column means the Security Functional Requirements dependency was included in the ST.

Table 24: TOE Security Functional Requirements Dependency Rationale

SFR	Dependencies	Rationale
FAU_ARP_ACK_(EXT).1	No Dependencies	Not applicable.
FAU_ARP.1	FAU_SAA.1	Met by FAU_SAA.1
FAU_GEN.1	FPT_STM.1	Met by FPT_STM.1
FAU_GEN.2	FAU_GEN.1	Met by FAU_GEN.1
	FIA_UID.1	Met by FAU_UID.2
FAU_SAA.1	No Dependencies	Not applicable.
FAU_SAR.1	FAU_GEN.1	Met by FAU_GEN.1
FAU_SAR.2	FAU_SAR.1	Met by FAU_SAR.1
FAU_SAR.3	FAU_SAR.1	Met by FAU_SAR.1
FAU_SEL.1	FAU_GEN.1	Met by FAU_GEN.1
	FMT_MTD.1	Met by FMT_MTD.1(1)
FAU_STG.1	FAU_GEN.1	Met by FAU_GEN.1
FAU_STG.3	FAU_STG.1	Met by FAU_STG.1
FAU_STG_(EXT).4	FAU_GEN.1	Met by FAU_GEN.1
FCS_BCM_(EXT).1	No Dependencies	Not applicable.
FCS_CKM.1(1)	FCS_CKM.2 or FCS_COP.1	Met by FCS_COP.1(1) Met by FCS_COP.1(2) Met by FCS_COP.1(3) Met by FCS_COP.1(4)
	FCS_CKM.4	Met by FCS_CKM.4
	FCS_COP_(EXT).1	Met by FCS_COP_(EXT).1
FCS_CKM.1(2)	FCS_CKM.2 or FCS_COP.1	Met by FCS_COP.1(1) Met by FCS_COP.1(2) Met by FCS_COP.1(3) Met by FCS_COP.1(4)

SFR	Dependencies	Rationale
	FCS_CKM.4	Met by FCS_CKM.4
FCS_CKM.2	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	Met by FCS_CKM.1(1) Met by FCS_CKM.1(2)
	FCS_CKM.4	Met by FCS_CKM.4
FCS_CKM_(EXT).2	No Dependencies	Not applicable.
FCS_CKM.4	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	Met by FCS_CKM.1(1) Met by FCS_CKM.1(2)
FCS_COP.1(1)	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	Met by FCS_CKM.1(1) Met by FCS_CKM.1(2)
	FCS_CKM.4	Met by FCS_CKM.4
FCS_COP.1(2)	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	Met by FCS_CKM.1(1) Met by FCS_CKM.1(2)
	FCS_CKM.4	Met by FCS_CKM.4
FCS_COP.1(3)	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	Met by FCS_CKM.1(1) Met by FCS_CKM.1(2)
	FCS_CKM.4	Met by FCS_CKM.4
FCS_COP.1(4)	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1	Met by FCS_CKM.1(1) Met by FCS_CKM.1(2)
	FCS_CKM.4	Met by FCS_CKM.4
FCS_COP_(EXT).1	No Dependencies	Not applicable.
FCS_IKE_(EXT).1	FCS_COP_(EXT).1	Met by FCS_COP_(EXT).1
FCS_GDOI_(EXT).1	FCS_CKM.2	Met by FCS_CKM.2
	FCS_COP.1	Met by FCS_COP.1(1)
FDP_IFC.1(1)	FDP_IFF.1	Met by FDP_IFF.1(1)
FDP_IFC.1(2)	FDP_IFF.1	Met by FDP_IFF.1(2)
FDP_IFC.1(4)	FDP_IFF.1	Met by FDP_IFF.1(4)
FDP_IFF.1(1)	FDP_IFC.1	Met by FDP_IFC.1(1)
	FMT_MSA.3	Met by FMT_MSA.3(1)
FDP_IFF.1(2)	FDP_IFC.1	Met by FDP_IFC.1(2)
	FMT_MSA.3	Met by FMT_MSA.3(2)
FDP_IFF.1(4)	FDP_IFC.1	Met by FDP_IFC.1(4)
	FMT_MSA.3	Met by FMT_MSA.3(2)
FDP_RIP.2	No Dependencies	Not applicable.
FIA_AFL.1	FIA_UAU.1	Met by FIA_UAU.1 Met by FIA_UAU.2
FIA_ATD.1(1)	No Dependencies	Not applicable.
FIA_ATD.1(2)	No Dependencies	Not applicable.
FIA_UAU.1	FIA_UID.1	Met by FIA_UID.2
FIA_UAU.2	FIA_UID.1	Met by FIA_UID.2
FIA_UAU_(EXT).5	No Dependencies	Not applicable.
FIA_UID.2	No Dependencies	Not applicable.
FIA_USB.1	FIA_ATD.1	Met by FIA_ATD.1(1)
FMT_MOF.1(1)	FMT_SMR.1	Met by FMT_SMR.2
	FMT_SMF.1	Met by FMT_SMF.1
FMT_MOF.1(2)	FMT_SMR.1	Met by FMT_SMR.2
	FMT_SMF.1	Met by FMT_SMF.1
FMT_MOF.1(3)	FMT_SMR.1	Met by FMT_SMR.2
	FMT_SMF.1	Met by FMT_SMF.1
FMT_MOF.1(4)	FMT_SMR.1	Met by FMT_SMR.2
	FMT_SMF.1	Met by FMT_SMF.1
FMT_MOF.1(5)	FMT_SMR.1	Met by FMT_SMR.2

SFR	Dependencies	Rationale
	FMT_SMF.1	Met by FMT_SMF.1
FMT_MOF.1(6)	FMT_SMR.1	Met by FMT_SMR.2
	FMT_SMF.1	Met by FMT_SMF.1
FMT_MOF.1(7)	FMT_SMR.1	Met by FMT_SMR.2
	FMT_SMF.1	Met by FMT_SMF.1
FMT_MOF.1(8)	FMT_SMR.1	Met by FMT_SMR.2
	FMT_SMF.1	Met by FMT_SMF.1
FMT_MSA.1(1)	FDP_ACC.1 or FDP_IFC.1	Met by FDP_IFC.1(2) Met by FDP_IFC.1(4)
	FMT_SMR.1	Met by FMT_SMR.2
	FMT_SMF.1	Met by FMT_SMF.1
FMT_MSA.1(2)	FDP_ACC.1 or FDP_IFC.1	Met by FDP_IFC.1(1)
	FMT_SMR.1	Met by FMT_SMR.2
	FMT_SMF.1	Met by FMT_SMF.1
FMT_MSA.1(3)	FDP_ACC.1 or FDP_IFC.1	Met by FDP_IFC.1(2) Met by FDP_IFC.1(4)
	FMT_SMR.1	Met by FMT_SMR.2
	FMT_SMF.1	Met by FMT_SMF.1
FMT_MSA.3(1)	FMT_MSA.1	Met by FMT_MSA.1(2)
	FMT_SMR.1	Met by FMT_SMR.2
FMT_MSA.3(2)	FMT_MSA.1	Met by FMT_MSA.1(1)
	FMT_SMR.1	Met by FMT_SMR.2
FMT_MSA.3(3)	FMT_MSA.1	Met by FMT_MSA.1(1)
	FMT_SMR.1	Met by FMT_SMR.2
FMT_MTD.1(1)	FMT_SMF.1	Met by FMT_SMF.1
	FMT_SMR.1	Met by FMT_SMR.2
FMT_MTD.1(2)	FMT_SMF.1	Met by FMT_SMF.1
	FMT_SMR.1	Met by FMT_SMR.2
FMT_MTD.1(3)	FMT_SMF.1	Met by FMT_SMF.1
	FMT_SMR.1	Met by FMT_SMR.2
FMT_MTD.1(4)	FMT_SMF.1	Met by FMT_SMF.1
	FMT_SMR.1	Met by FMT_SMR.2
FMT_MTD.2(1)	FMT_MTD.1	Met by FMT_MTD.1(1)
	FMT_SMR.1	Met by FMT_SMR.2
FMT_MTD.2(2)	FMT_MTD.1	Met by FMT_MTD.1(1)
	FMT_SMR.1	Met by FMT_SMR.2
FMT_MTD.2(3)	FMT_MTD.1	Met by FMT_MTD.1(1)
	FMT_SMR.1	Met by FMT_SMR.2
FMT_REV.1	FMT_SMR.1	Met by FMT_SMR.2
FMT_SMF.1	No Dependencies	Not applicable.
FMT_SMR.2	FIA_UID.1	Met by FIA_UID.2
FPT_FLS.1	No Dependencies	Not applicable.
FPT_HA_(EXT).1	No Dependencies	Not applicable.
FPT_ITA.1	No Dependencies	Not applicable.
FPT_ITC.1	No Dependencies	Not applicable.
FPT_ITI.1	No Dependencies	Not applicable.
FPT_RCV.1	AGD_OPE.1	Met by AGD_OPE.1
FPT_RCV.2	AGD_OPE.1	Met by AGD_OPE.1
FPT_RPL.1	No Dependencies	Not applicable.
FPT_STM.1	No Dependencies	Not applicable.
FPT_PRO_(EXT).1	No Dependencies	Not applicable.

SFR	Dependencies	Rationale
FPT_TDC.1	No Dependencies	Not applicable.
FPT_TST.1(1)	No Dependencies	Not applicable.
FPT_TST.1(2)	No Dependencies	Not applicable.
FPT_TST_(EXT).1	No Dependencies	Not applicable.
FRU_RSA.1(1)	No Dependencies	Not applicable.
FRU_RSA.1(2)	No Dependencies	Not applicable.
FRU_RSA.1(3)	No Dependencies	Not applicable.
FTA_SSL.1	FIA_UAU.1	Met by FIA_UAU.2
FTA_SSL.2	FIA_UAU.1	Met by FIA_UAU.2
FTA_SSL.3	No Dependencies	Not applicable.
FTA_TAB.1	No Dependencies	Not applicable.
FTA_TSE.1	No Dependencies	Not applicable.
FTP_ITC.1(1)	No Dependencies	Not applicable.
FTP_ITC.1(2)	No Dependencies	Not applicable.
FTP_TRP.1(1)	No Dependencies	Not applicable.
FTP_TRP.1(2)	No Dependencies	Not applicable.

5.5 Security Assurance Requirements

5.5.1 SAR Requirements

The TOE assurance requirements for this ST is EAL4 Augmented ALC_FLR.2 derived from Common Criteria Version 3.1, Revision 3. The Security Target Claims conformance to EAL4 Augmented with ALC_FLR.2. The assurance requirements are summarized in the table below, with the extended requirements in **bold** print.

Table 25: Assurance Measures

Assurance Class	Components	Components Description
DEVELOPMENT	ADV_ARC.1	Security Architectural Description
	ADV_FSP.4	Complete functional specification
	ADV_IMP.1	Implementation of the TSF
	ADV_TDS.3	Basic modular design
GUIDANCE DOCUMENTS	AGD_OPE.1	Operational user guidance
	AGD_PRE.1	Preparative User guidance
LIFE CYCLE SUPPORT	ALC_CMC.4	Product support, acceptance procedures and automation
	ALC_CMS.4	Problem tracking CM coverage
	ALC_DEL.1	Delivery procedures
	ALC_DVS.1	Identification of security measures
	ALC_FLR.2	**Flaw Reporting Procedures**
	ALC_LCD.1	Developer defined life-cycle model
	ALC_TAT.1	Well-defined development tools
TESTS	ATE_COV.2	Analysis of coverage
	ATE_DPT.1	Testing: modular design
	ATE_FUN.1	Functional testing
	ATE_IND.2	Independent testing - sample
VULNERABILITY ASSESSMENT	AVA_VAN.3	Focused vulnerability analysis

5.5.2 Security Assurance Requirements Rationale

This Security Target claims conformance to EAL4 Augmented with ALC_FLR.2. This target was chosen to ensure that the TOE has a moderate level of assurance in enforcing its security functions when instantiated in its intended environment which imposes no restrictions on assumed activity on applicable networks. Augmentation was chosen to address having flaw remediation procedures and correcting security flaws as they are reported.

5.6 Assurance Measures

The TOE satisfies the identified assurance requirements. This section identifies the Assurance Measures applied by Cisco to satisfy the assurance requirements. The table below lists the details.

Table 26: Assurance Measures

Component	How requirement will be met
ADV_ARC.1	The architecture of the TOE that is used to protect the TSF documented by Cisco in their development evidence.
ADV_FSP.4	The externally visible interfaces of the TOE used by the users of the TOE along with the description of the security functions and a correspondence between the interfaces and the security functions from the ST are documented by Cisco in their development evidence. The development evidence also contains a tracing to the SFRs described in this ST.
ADV_IMP.1	Cisco provides access to the TSF implementation to the evaluation lab.
ADV_TDS.3	The design of the TOE will be described in the development evidence. This evidence will also contain a tracing to the TSFI defined in the FSP.
AGD_OPE.1	The administrative guidance is detailed to provide descriptions of how administrative users of the TOE can securely administer the TOE using those functions and interfaces detailed in the guidance.
AGD_PRE.1	Cisco documents the installation, generation, and startup procedures so that the users of the TOE can put the components of the TOE in the evaluated configuration.
ALC_CMC.4	Cisco performs configuration management on configuration items of the TOE. Configuration management is performed on the TOE and the implementation representation of the TOE.
ALC_CMS.4	Cisco uniquely identifies configuration items and each release of the TOE has a unique reference. The Configuration Management documentation contains a configuration item list.
ALC_DEL.1	Cisco documents the delivery procedure for the TOE to include the procedure on how to download certain components of the TOE from the Cisco website and how certain components of the TOE are physically delivered to the user. The delivery procedure detail how the end-user may determine if they have the TOE and if the integrity of the TOE has been maintained. Further, the delivery documentation describes how to acquire the proper license keys to use the TOE components.
ALC_DVS.1	Cisco implements security controls over the development environment. Cisco meets these requirements by documenting the security controls.
ALC_FLR.2	Cisco documents the flaw remediation and reporting procedures so that security flaw reports from TOE users can be appropriately acted upon, and TOE users can understand how to submit security flaw reports to the developer.
ALC_LCD.1	Cisco documents the TOE development life-cycle to meet these requirements.
ALC_TAT.1	Cisco uses well-defined development tools for creating the TOE.
ATE_COV.2	Cisco demonstrates the interfaces tested during functional testing using a coverage analysis.
ATE_DPT.1	Cisco demonstrates the TSF subsystems tested during functional testing using a depth analysis.
ATE_FUN.1	Cisco functional testing documentation contains a test plan, a description of the tests, along with the expected and actual results of the test conducted against the functions specified in the ST.
ATE_IND.2	Cisco will help meet the independent testing by providing the TOE to the evaluation facility.

Component	How requirement will be met
AVA_VAN.3	Cisco will provide the TOE for testing.

6 TOE SUMMARY SPECIFICATION

6.1 TOE Security Functional Requirement Measures

This chapter identifies and describes how the Security Functional Requirements identified above are met by the TOE.

Table 27: How TOE SFRs Measures

TOE SFRs	How the SFR is Met
FAU_ARP.1	The TOE supports a local, directly connected (or connected through a terminal server), management interface and remote management through a SSHv2 secure tunnel. A security alarm is immediately displayed and the contents of the audit record that triggered the alarm is accessible to both the local and remote administrator whenever a potential security violation is detected by the TOE. The TOE also provides the facility for the Security Administrator to enable an auditable alarm that sounds for both the local and remote administrator through the TOE administrative CLI. This audible alarm sounds until it is acknowledged by an administrative user. These alarms are stored in the same internal syslog server as other audit events. Each alarm contains the following information, the date and time the alarm was generated, the type of event that caused the alarm, and where the event that caused the alarm took place.
FAU_ARP_ACK_(EXT).1	The TOE displays a security alarm to both the local and remote administrator each time a potential security violation is identified. Additionally, the TOE supports an auditable alarm that may be configured to sound by the Security Administrator. These alarms are displayed/sound until they are acknowledged by an administrator. After the alarm is acknowledged through the administrative CLI, a confirmation message is displayed. These alarms are stored in the same internal syslog server as other audit events. Each alarm contains the following information, the date and time the alarm was generated, the type of event that caused the alarm, and where the event that caused the alarm took place.
FAU_GEN.1	The TOE generates an audit record that is stored internally within the TOE whenever an auditable event occurs. The types of events that cause audit records to be generated include, audit searching and sorting related events, cryptography related events, events related to the enforcement of information flow policies, identification and authentication related events, and administrative events (the specific events and the contents of each audit record are listed in the table within the FAU_GEN.1 SFR, "Auditable Events Table"). Each of the events is specified in the syslog internal to the TOE in enough detail to identify the user for which the event is associated, when the event occurred, where the event occurred, the outcome of the event, and the type of event that occurred. Note that the audit functionality is enabled by default in the evaluated configuration and cannot be disabled.
FAU_GEN.2	The syslog entry associated with each auditable event generated by the TOE is described in sufficient detail to allow an administrative user reviewing the audit log to identify which user the auditable event is associated with.
FAU_SAA.1	The TOE internally tracks the number of occurrences associated with specific types of auditable event (for example, authentication failures or information flow policy violations). Once the administratively configured threshold of a specific event type is met, the TOE identifies the event as a security violation. This happens regardless of the length of time that has lapsed between failed authentication attempts. This functionality is configured through the TOE CLI by the Security Administrator. The specific events that can be monitored include: • Security Administrator specified number of authentication failures;

TOE SFRs	How the SFR is Met
	• Security Administrator specified number of Information Flow policy violations by an individual presumed source network identifier (e.g., IP address) within an administrator specified time period; • Security Administrator specified number of Information Flow policy violations to an individual destination network identifier within an administrator specified time period; • Security Administrator specified number of Information Flow policy violations to an individual destination subject service identifier within an administrator specified time period; • Security Administrator specified Information Flow policy rule, or group of rule violations within an administrator specified time period; • Any detected replay of TSF data or security attributes; • Any failure of the cryptomodule/cryptographic self-tests; • Any failure of the other key generation self-tests; • Any failure of the other TSF self-tests; • Security Administrator specified number of encryption failures; • Security Administrator specified number of decryption failures; • Security Administrator specified number of Phase 1 authentication failures when negotiating the Internet Key Exchange protocol; • Security Administrator specified number of failures occur during Phase 2 negotiation;
FAU_SAR.1	The TOE provides the ability for the administrators of the TOE to view all audit events stored within the TOE. The TOE provides CLI commands that allow an administrative user to display the audit event to the console screen.
FAU_SAR.2	The TOE controls the access available to an administrative user to only the resource associated with the user's role. The TOE does not provide any methods to access the internally stored audit records. All attempts to view the audit data stored within the TOE are mediated by the TOE itself. Only administrative users of the TOE are able to access the audit records stored within the TOE.
FAU_SAR.3	Through the TOE CLI administrative interface, the TOE provides the ability for authorized administrative users to search and sort the internally stored audit records. The TOE provides dedicated CLI to all Administrators to facilitate search and sorting of audit records within the TOE. The criteria for which audit records can be searched and sorted include, user identity, source subject identity, destination subject identity, ranges of one or more: (dates, times, user identities, subject service identifiers, or transport layer protocol), Rule identity, TOE network interfaces.
FAU_SEL.1	The TOE provides the ability to include or exclude the types of auditable events tracked by the TOE. This capability is provided through the TOE CLI administrative interface available to the Security Administrator. After configured by a Security administrator, the TOE will exclude specific auditable events. Otherwise, all types of auditable events will be included in the audit records. The criteria that are configurable by the Security Administrator include: user identity, event type, network identifier (the IP address or "console" identification), subject service identifier (type of service), success of auditable security events, failure of auditable security events, and rule identity. When an audit event is excluded, the event is generated but not stored in TOE persistent memory.
FAU_STG.1	Through the TOE CLI administrative interface, the TOE provides the ability for authorized administrative users (audit administrator) to delete audit records stored within the TOE. The TOE provides dedicated CLI commands that are only available to the Audit Administrator to facilitate the deletion of audit records.
FAU_STG_(EXT).4	The TOE monitors the amount of free storage space available for audit records

TOE SFRs	How the SFR is Met
	stored internal to the TOE. After the storage space available for audit records is used to a specific level, the TOE takes specific actions as configured by the Security Administrator. The possible actions taken by the TOE include stopping the TOE to prevent further audit events until the TOE is restarted by an authorized administrator or overwriting the oldest audit records with the newest audit records.
FAU_STG.3	The TOE immediately displays a message to the TOE administrative users (both local and remote) whenever the Security Administrator configured audit storage threshold is met. This threshold is configured through the TOE provided CLI administrative interface by the Security Administrator. Additionally, the TOE will send an auditable alarm to the administrator work station, if the TOE is configured to send auditable alarms.
FCS_BCM_(EXT).1	The cryptography provided by the TOE has been FIPS 140-2 validated to overall level 2 with sections 3, 7, and 10 validated to level 3. Please see FIPS certificate # 1390 for validation details.
FCS_CKM.1(1)	Symmetric cryptographic keys used within the TOE for IPSec and SSHv2 secure management are generated using the implemented FIPS-approved ANSI X9.31 PRNG. The TOE key generation capabilities have been verified as part of the TOE's FIPS 140-2 validation. The symmetric keys used for IPSec are created through the IKE protocol interaction. The keys are transient and are protected in transit by a TCP protocol CRC. The symmetric keys used for SSHv2 are created through the SSHv2 protocol interaction. The keys are transient and are protected by the security mechanisms present in the SSHv2 protocol. Further details regarding the FIPS validation can be found in Certificate #1390, Cisco ASR1002 Series Router, Cisco ASR1004 Series Router, Cisco ASR1006 Series Router.
FCS_CKM.1(2)	Asymmetric cryptographic keys used within the TOE for IPSec and SSHv2 secure management are generated using the implemented FIPS-approved ANSI X9.31 PRNG. The TOE key generation capabilities have been verified as part of the TOE's FIPS 140-2 validation. Asymmetric key integrity is verified by performing a pair-wise key consistency check required by FIPS. Further details regarding the FIPS validation can be found in Certificate #1390, Cisco ASR1002 Series Router, Cisco ASR1004 Series Router, Cisco ASR1006 Series Router.
FCS_CKM.2	When a generated key pair is not explicitly designated as exportable a signing private key is not extractable from the TOE and may not be distributed to other entities. A generated key pair designated for export may be retrieved from TOE in a cryptographically protected format only. This functionality was verified as part of the TOE FIPS 140-2 validation. A key pair may be imported manually by cutting and pasting its ASCII-hex representation, imported in a PEM or PKCS#12 bundle. Upon key pair import it is verified for consistency. This is a mandatory FIPS requirement. The key used to encrypt administrative authentication credentials may be entered through the administrative CLI wither through a directly connected console or over an encrypted SSHv2 administrative session. Further details regarding the FIPS validation can be found in Certificate #1390, Cisco ASR1002 Series Router, Cisco ASR1004 Series Router, Cisco ASR1006 Series Router. NOTE: In support of NIST SP 800-56, the TOE supports DH primitive according to Section 5.7.1.1 of FIPS SP800-56A. The TOE supports PKCS#7 wrapped certificates for ISAKMP.
FCS_CKM_(EXT).2	After IPsec or SSHv2 encryption and authentication keys are generated within the TOE they are transferred into the cryptographic coprocessor within the TOE. The keys are packaged into a message which is moved internally within the TOE over a TCP/IP based reliable IPC mechanism. TCP/IP provides a built-in mechanism for checking message payload integrity. The Security Administrator has the ability to limit the lifetime of inactive

TOE SFRs	How the SFR is Met
	symmetric cryptographic keys used in IPsec security associations (SA) by configuring the IPsec SA lifetime, the Dead Peer Detection or the Idle Peer Detection IPsec SA properties. The TOE also provides the Security Administrator the ability to limit the lifetime of symmetric cryptographic keys used in IKE SA by specifying their maximal lifetime. Idle IKE SA's are destroyed when the lifetime expires. A Security Administrator enforces lifetime of asymmetric cryptographic key pairs associated with a digital certificate by specifying certificate validity period when requesting a certificate from a Certificate Authority during the certificate enrollment procedure. When a certificate and a corresponding private key are imported in a protected cryptographic bundle into TOE certificate validity is verified including the certificate expiration date. The import operation is rejected if the certificate has expired. This functionality was verified as part of the TOE FIPS 140-2 validation. Further details regarding the FIPS validation can be found in Certificate #1390, Cisco ASR1002 Series Router, Cisco ASR1004 Series Router, Cisco ASR1006 Series Router.
FCS_CKM.4	The TOE zeroizes all of the cryptographic keys used within the TOE after the key is no longer of use to the TOE. The key and CSP zeroization capabilities of the TOE have been verified as part of the TOE's FIPS 140-2 validation. Further details regarding the FIPS validation can be found in Certificate #1390, Cisco ASR1002 Series Router, Cisco ASR1004 Series Router, Cisco ASR1006 Series Router.
FCS_COP.1(1)	The TOE provides AES encryption and decryption in support of IPSec tunneling and SSHv2 secure management. AES data encryption (128-bit, 196-bit, and 256-bit) is the encryption/decryption option that is used within SSHv2 communications with the TOE. Specifically, AES is used to encrypt the following traffic, IKE Session traffic, IPSec session traffic, SSHv2 session traffic, and GDOI traffic. The assymmetric encryption and decryption as used in these protocols was evaluated as part of the TOE's FIPS 140-2 validation. Additionally, AES can optionally be chosen by the administrator to encrypt stored administrative authentication credentials. Further details regarding the FIPS validation can be found in Certificate #1390, Cisco ASR1002 Series Router, Cisco ASR1004 Series Router, Cisco ASR1006 Series Router.
FCS_COP.1(2)	The TOE provides cryptographic signatures in support of IPSec tunneling and SSHv2 secure management. The TOE provides the RSA option in support of SSHv2 key establishment. RSA (3072-bit and 4096-bit) is used in the establishment of IPSec and SSHv2 key establishment. For SSHv2, RSA host keys are supported. The symmetric encryption and decryption as used in these protocols was evaluated as part of the TOE's FIPS 140-2 validation. Further details regarding the FIPS validation can be found in Certificate #1390, Cisco ASR1002 Series Router, Cisco ASR1004 Series Router, Cisco ASR1006 Series Router.
FCS_COP.1(3)	The TOE provides SHS hashing in support of IPSec tunneling and SSHv2 secure management. The TOE provides the SHS hashing option in support of SSHv2 key establishment. SHS hashing (SHA-1) is used in the establishment of IKE sessions, IPSec sessions, and SSHv2 sessions. The hashing used in these protocols was evaluated as part of the TOE's FIPS 140-2 validation. Further details regarding the FIPS validation can be found in Certificate #1390, Cisco ASR1002 Series Router, Cisco ASR1004 Series Router, Cisco ASR1006 Series Router.
FCS_COP.1(4)	The TOE provides Diffie-Hellman Key Exchange services in support of IPSec tunneling and SSHv2 secure management. The TOE provides the Diffie-Hellman key agreement (groups 1, 14, and 16) option in support of SSHv2 key establishment. This is used to support both the transport and user authentication

TOE SFRs	How the SFR is Met
	protocols within SSH. The key agreement used in these protocols was evaluated as part of the TOE's FIPS 140-2 validation. Further details regarding the FIPS validation can be found in Certificate #1390, Cisco ASR1002 Series Router, Cisco ASR1004 Series Router, Cisco ASR1006 Series Router. During the FIPS 140-2 testing, the source code associated with the Diffie-Hellman Key Exchange is verified for correctness. Also, the TOE implements a ModExp POST. This self test was verified during the FIPS 140-2 validation testing. NOTE: In support of NIST SP 800-56, supports FFC DH primitive according to Section 5.7.1.1 of FIPS SP800-56A. The TOE supports PKCS#7 wrapped certificates for ISAKMP.
FCS_COP_(EXT).1	In support of the provided cryptography, the TOE implements a pseudo Random Number Generator. This PRNG that is implemented is a FIPS-approved 3-key TDES based ANSI X9.31 compliant PRNG seeded from both a hardware and software entropy source. The TSF prevents tampering of the seeding entropy sources through the FIPS 140-2 physical security mechanisms. FIPS 140-2 includes physical security requirements that prevent the tampering of any cryptographic key or seeding material stored within the cryptographic module. Additionally, all entropy is only stored internally to the cryptographic module in DRAM and cannot be accessed internally. The CSTL performing the FIPS 140-2 validation verified that there is no way to access entropy as part of the cryptographic module testing. The cryptographic services supported with this RNG include IPSec key establishment and SSHv2 key establishment. This service was evaluated as part of the TOE's FIPS 140-2 validation. Further details regarding the FIPS validation can be found in Certificate #1390, Cisco ASR1002 Series Router, Cisco ASR1004 Series Router, Cisco ASR1006 Series Router.
FCS_IKE_(EXT).1	The TOE provides the cryptographic services necessary to support IPSec connections with remote IT entities wishing to pass information through an IPSec protected tunnel. The TOE fully supports Internet Key Exchange (IKE), RFC 2409, as follows: • Phase 1, the establishment of a secure authenticated channel between the TOE and another remote VPN endpoint, shall be performed using one of the following, as configured by the security administrator, Main Mode, Aggressive Mode • Phase 2, negotiation of security services for IPsec, shall be done using Quick Mode, using SHA-1 as the pseudo-random function. Quick Mode shall generate key material that provides perfect forward secrecy. The use of SHA-256 and SHA-384 as the PRF in IKEv1 KDF although allowed by the requirements , is not supported. • x of g^xy is randomly generated using a FIPS-approved random number generator • The minimum size of x is twice the number of bits of the strength level associated with the negotiated DH group • The nonce sizes are between 8 and 256 bytes. • Nonces are generated in a manner such that the probability that a specific nounce value will be repeated during the life a specific IPsec SA is less than 1 in 2^(bit strength of the negotiated DH group) • Preshared key are generated using the FIPS approved random number generator • The TSF computes the value of SKEYID (as defined in RFC 2409), using SHA-1 as the pseudo-random function • The following authentication methods are supported: o SKEYID_d = prf(SKEYID, g^xy | CKY-I | CKY-R | 0) o SKEYID_a = prf(SKEYID, SKEYID_d | g^xy | CKY-I | CKY-

TOE SFRs	How the SFR is Met
	R | 1) o SKEYID_e = prf(SKEYID, SKEYID_a | g^xy | CKY-I | CKY-R | 2) • When authenticating a Phase 1 exchange, the TSF generates HASH_I if it is the initiator, or HASH_R if it is the responder as follows o HASH_I = prf(SKEYID, g^xi | g^xr | CKY-I | CKY-R | SAi_b | IDii_b) o HASH_R = prf(SKEYID, g^xr | g^xi | CKY-R | CKY-I | SAi_b | IDir_b) • The TSF is capable of authenticating IKE Phase 1 using the following methods o The TSF can use RSA digital signature o when an RSA signature is applied to HASH I or HASH R it is PKCS#1 encoded. The TSF checks the HASH_I and HASH_R values sent against a computed value to detect any changes made to the proposed transform negotiated in phase one. If changes are detected the session is terminated and an alarm generated. o For X.509 V3 certificates, the TOE is capable of checking for validity of the certificate path, and at option of SA, check for certificate revocation. o The TSF supports authentication using a pre-shared key. • The TSF computes the hash values for Quick Mode in the following way: o HASH(1) = prf(SKEYID_a, M-ID |[any ISAKMP payload after o HASH(1) header contained in the message)] o HASH(2) = prf(SKEYID_a, M-ID | Ni_b | [any ISAKMP payload after HASH(2) header contained in the message)] o HASH(3) = prf(SKEYID_a, 0 | M-ID | Ni_b | Nr_b) • The TSF computes new keying material during Quick Mode as follows: o when using perfect forward secrecy - KEYMAT = prf(SKEYID_d, g(qm)^xy | protocol | SPI | Ni_b | Nr_b), o When perfect forward secrecy is not used - KEYMAT = prf(SKEYID_d | protocol | SPI | Ni_b | Nr_b) • The TSF supports the following ID types: ID_IPV4_ADDR, ID_IPV6_ADDR, ID_FQDN, ID_USER_FQDN, [ID_IPV4_ADDR_SUBNET, ID_IPV6_ADDR_SUBNET, ID_IPV4_ADDR_RANGE, ID_IPV6_ADDR_RANGE, ID_DER_ASN1_DN, ID_DER_ASN1_GN, ID_KEY_ID The TOE supports Diffie-Hellman groups 15 (3072 bit keys) and 16 (4096 bit keys). This is consistent with the requirements specified in the Protection Profiles for which conformance is claimed. Further details regarding the FIPS validation can be found in Certificate #1390, Cisco ASR1002 Series Router, Cisco ASR1004 Series Router, Cisco ASR1006 Series Router.
FCS_GDOI_(EXT).1	In support of IPSec the TOE provides a key transport method of a key server transferring cryptographic keys and policy to authenticated and authorized group members over Internet Protocol. The TOE supports GDOI, RFC 3547. The TSF supports “GROUPKEY PUSH” and “GROUPKEY PULL” for keying and rekeying. This service was evaluated as part of the TOE’s FIPS 140-2 validation. Further details regarding the FIPS validation can be found in Certificate #1390, Cisco ASR1002 Series Router, Cisco ASR1004 Series Router, Cisco ASR1006 Series Router.
FDP_IFC.1(1)	The TOE facilitates VPN connections with other IPSec capable IT entities. The TOE first determines if the communication is allowed. After it is determined that

TOE SFRs	How the SFR is Met
	the VPN connection is allowed, the TOE participates in the IPSec communication based on the established IPSec parameters. When network packets are received on a TOE interface, the TOE verifies whether the packet is allowed or not and performs one of the following actions, pass packets to the destination without modifying; send IPSec encrypted and authenticated packets to a peer TOE using ESP in tunnel mode as defined in RFC 2406; decrypt and verify authentication and pass received packets from a peer TOE in tunnel mode using ESP.
FDP_IFC.1(2)	The TOE allows access to some TOE provided services prior to authentication. All traffic attempting to use TOE provided services are mediated by the TOE itself. The decisions to allow access to are made by the TOE for each access attempt. When network packets are received on a TOE interface that are destined to the TOE, the TOE verifies whether the packet is allowed or not and performs one of the following actions: accept the network packets; reject the network packets.

FDP_IFC.1(4)	The TOE enforces information flow policies on traffic through the TOE from unauthenticated IT entities. These policies are enforced on network packets that are receive by TOE interfaces and leave the TOE through other TOE interfaces. When network packets are received on a TOE interface from an unauthenticated source, the TOE verifies whether the network traffic is allowed or not and performs one of the following actions,
	• pass information by opening a relay connection through the TSF on behalf of the source subject to the destination subject, and with the TSF ensuring the following conditions:
	  ○ the connection from the source subject is from a valid peer network,
	  ○ the new relay connection is established to the destination subject on a valid peer network.
	• pass information

FDP_IFF.1(1)	The TOE facilitates IPSec VPN communication with IPSec enabled IT devices. The TOE compares plaintext traffic received from IPSec VPN or destined to IPsec VPN to the configured information flow policies. If the information flow meets a configured information flow policy that allows the traffic, then traffic originated from a VPN tunnel or destined to a VPN tunnel is permitted. If the information flow meets a configured policy that denies traffic, such traffic is not permitted. The TOE allows network traffic for the following scenarios: • 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 subject 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. The TOE examines a packet's source IP address, destination IP address, transport protocol, and layer 4 source and destination ports and compares them to the configured VPN policy to determine the action to apply to the network packets. If the packet is a plaintext packet that matches a policy rule that allows packets to be passed without modification, the packet is passed without modification. If the packet is a plaintext packet that matches a policy rule that requires the TOE to send IPSec encrypted and authenticated packets to a peer, the TOE encrypts and applies a authentication mechanism to the packet using ESP in tunnel mode as defined in RFC 2406 and sends it to its peer. If the packet matches a policy that requires the TOE to decrypt, verify authentication and pass

TOE SFRs	How the SFR is Met
	received packets from a peer TOE in tunnel mode using ESP, the TOE decrypts, verifies authentication and passes received packets from a peer TOE in tunnel mode using ESP and • the selected information flow policy rule specifies that the information flow is to be permitted The TOE denies network traffic for the following scenarios: • The TOE rejects requests for access or services when the traffic is received from an IP or MAC address that is not included in the set of allowed addresses; • The TOE shall reject requests for access or services when the traffic is received from an IP or MAC address that is a broadcast identity; • The TOE shall reject requests for access or services when the traffic is received from an IP address that is defined as a loopback address; • The TOE shall reject requests in which the information received by the TOE contains the route by which the network traffic must travel Additionally, the TOE provides the Security Administrator with the capability to view all information flows allowed by the information flow policy ruleset before the policy is applied.
FDP_IFF.1(2)	The Security Administrator configures unauthenticated TOE Service policies for network traffic requesting services provided by the TOE. These information flow policies consist of a zone pair describing from where traffic is initiated to where traffic is destined, and description of the operation (whether the traffic is allowed or not allowed through the zone pair), and the type of traffic for which the policy is applicable (Source IP address, destination IP address, ICMP code, and ICMP message type). A zone is a configurable group of TOE interfaces for which the policies are applied. Traffic destined to the TOE itself is destined for a predefined zone known as the "self" zone. Before the policies are applied to the information flow policies, the Security Administrator can review the policies through the TOE CLI. When network traffic is received by the TOE requesting TOE services (i.e., ICMP), the source of the traffic (IP address) and the attributes of the packet are compared against the policy together through TCAM lookup. The TOE then allows or does not allow the traffic to flow depending on the information flow policy for which the traffic meets. In the following instances, the TOE rejects the information flow: • The TOE rejects requests for access or services where the source IP address of the network traffic is not included in the set of allowed source IP address; • The TOE rejects requests for access or services where the source IP address of the network traffic is a broadcast address; • The TOE rejects requests for access or services where the source IP address of the network traffic is a defined loopback address; and • The TOE rejects requests for which the network traffic received by the TOE that specifies the route by which the traffic flows The first packet in the flow is checked against zonepair+policy, then a session is created, then the following packets on the flow in both directions are allowed by the session. There is no need to have zonepair+policy on the other direction for this flow. Note that this does not apply to ICMP packets, since ICMP has no sessions. Session implies state and, as such, is more that one exchange. A single request-reply exchange does not constitute a session.

TOE SFRs	How the SFR is Met
	Additionally, the TOE provides the Security Administrator with the capability to view all information flows allowed by the information flow policy ruleset before the policy is applied.
FDP_IFF.1(4)	The Security Administrator configures unauthenticated information flow policies for network traffic flowing through the TOE. These information flow policies consist of a zone pair describing from where traffic is initiated to where traffic is destined, and description of the operation (whether the traffic is allowed or not allowed through the zone pair), and the type of traffic for which the policy is applicable (Source IP address, destination IP address, and transport layer protocol, source and destination port number. A zone is a configurable group of TOE interfaces for which the policies are applied. When network traffic is received the TOE, the TOE examines the attributes of the packet and compares the traffic to the configured information flow policies. The TOE finally allows or does not allow the traffic to flow depending on the information flow policy for which the traffic meets. The TOE also identifies the state of the traffic by inspecting the packets' sequence number, acknowledgement number, and flags for connection-oriented protocols, or by examining the source and destination address and traffic protocol for connectionless traffic. If the traffic is part of an established session, the traffic is allowed. Otherwise, the TOE allows or denies traffic to flow depending on the information flow policy for which the traffic meets. For messages received in fragments, the TOE also reassembles the packets using Virtual Fragment Reassembly. The TOE holds all fragments it receives until it receives a full message and assembles them. If the assembled message is permitted to flow, then the TOE passes the packet. The traffic is sent out as it was received, fragmented. The first packet in the flow is checked against zonepair+policy, then a session is created, then the following packets on the flow in both directions are allowed by the session. There is no need to have zonepair+policy on the other direction for this flow. Note that this does not apply to ICMP. A single request-reply exchange does not constitute a session. Additionally, the TOE provides the Security Administrator with the capability to view all information flows allowed by the information flow policy ruleset before the policy is applied.
FDP_RIP.2	The TOE ensures that packets transmitted from the TOE do not contain residual information from previous packets. Packets that are not the required length use a four-byte repeating pattern for padding. Residual data is never transmitted from the TOE. Once packet handling is completed its content is overwritten before memory buffer which previously contained the packet is reused. This applies to both data plane traffic and administrative session traffic. Additionally, the TOE ensures that there is no reuse among administrative sessions. The administrative roles are defined so that there is not overlapping administrative controls between administrators with different roles. Since, administrators of different roles do not act upon the same administrative objects there cannot be reuse between administrative sessions of separate roles. Each administrative session takes place either through a directly connected cable or over an encrypted remote session that is distinct from all other encrypted

TOE SFRs	How the SFR is Met
	administrative sessions.
FIA_AFL.1	The TOE provides the ability for the Security Administrator to configure the actions the TOE takes when a Security Administrator configured number of unsuccessful authentication attempts occur. The TOE can be configured either to prevent remote authentication for a specific period of time or disallow remote administration of the TOE until the Security Administrator has re-enabled remote authentication. All remote administration is via SSHv2. Note: Local authentication is always allowed. VPN peers are not locked out by automated mechanisms. The IKEv1 protocol provides a pre-shared key method of an ISAKMP SA establishment, and when this method is used any IKE peer which possesses a pre-shared secret key is considered legitimate due to the anonymous nature of the IKEv1 DH key exchange procedure. Thus, policy based VPN peer lockout can only be achieved by manual methods (e.g. a pre-shared key removal or modification).
FIA_ATD.1(1)	The TOE internally maintains a profile for each administrator authorized to use the TOE. For each administrator, the TOE maintains, the username associated with the user, the user's password, and the roles supported by the TOE. The user name and password are used by the TOE to authenticate an administrator wishing to gain access to the TOE management functionality. The role is used by the TOE to allow an authenticated user to assume a predefined TOE role.
FIA_ATD.1(2)	The TOE internally maintains a profile containing the identity (IP address/host name) and the IKE security attributes for each VPN Peer (identified as authorized subjects within the SFR) for which the TOE communicates.
FIA_UAU.1	The TOE provides the ability for an authorized administrator to configure all TOE services through the TOE administrative CLI. When a TOE interface is configured and the "no shut" command is applied it responds to the "to us" pings but passing any other traffic requires configuring an appropriate route.
FIA_UAU.2	Administrative access to the TOE is facilitated through the TOE provided CLI. The TOE mediates all actions through the CLI. Once a potential administrative user attempts to access the management functionality of the TOE through either a directly connected console or remotely through an SSHv2 connection, the TOE prompts the user for a user name and password. Only after the administrative user presents the correct authentication credentials will access to the TOE administrative functionality be granted. No access is allowed to the administrative facilities of the TOE until an administrator is authenticated.
FIA_UAU_(EXT).5	The TOE locally maintains authentication credentials. All authentication provided by the TOE is accomplished by comparing the supplied credentials to the locally stored credentials within the TOE. The User passwords may be stored in plaintext, hashed or encrypted with AES encryption depending on the options set when username is created.
FIA_UID.2	All users wishing to use TOE services are identified prior to being allowed access to any of the services. Once a user attempts to access the management functionality of the TOE, the TOE prompts the user for a user name and password. Only after the administrative user presents the correct identification and authentication credentials will access to the TOE functionality be granted.
FIA_USB.1	The TOE is able to associate the administrative users and VPN Peers with their identifying attributes stored internally to the TOE. When associating the administrative user/remote IT entities with the associated security attributes the TOE applies the following rules, • Remote IT Entity (VPN Peers/VPN Gateways): Whenever the TOE negotiates an IPSec connection with a VPN Peer, the TOE compares the IKE Secure attributes to the internally store peer profile and associates the peer with the profile; • User: Whenever a user presents authentication credentials to the TOE via

TOE SFRs	How the SFR is Met
	the TOE CLI, the TOE verifies that the user is a known user and verifies that the user is associated with the desired role. The TOE associates the user with the profile within the TOE. The TOE then associates the role permissions with the user. If a change is required to the security attributes associated with the user/remote IT entities, the following rules are applied: • Remote IT Entity (VPN Peers/VPN Gateways): the security attributes (Subject identity/ IKE Security Attributes) can be changed by renegotiating a IPSec connection with the remote IT entity or by the TOE administrative user updating Remote IT entity profile within the TOE; • User: there are no 'users' on the TOE. All administrative credentials may be changed only during setup (not when the TOE is operational).
FMT_MOF.1(1)	There is no functionality provided for altering the frequency at which the image integrity test is run. It is executed at every reload of the TOE, which is available to all administrators.
FMT_MOF.1(2)	Cryptographic power up self-tests are mandatory for FIPS 140-2 validation and cannot be disabled. Once the ASR has powered up, the Cryptographic Administrator may invoke cryptographic self-tests on demand or run those periodically through the provided CLI interface.
FMT_MOF.1(3)	The TOE provides the ability to enable, disable, determine, and modify the behavior of the TOE audit review functionality through the provided CLI interface. All Administrators can view audit information.
FMT_MOF.1(4)	The TOE provides the ability to enable, disable, determine, and modify the behavior of the TOE security audit analysis behavior (including selecting the events that are audited and identifying potential security violations) through the provided CLI interface. This functionality is limited to the Security Administrator.
FMT_MOF.1(5)	The TOE provides the ability to enable or disable the TOE security alarms behavior through the provided CLI interface. This functionality is limited to the Security Administrator.
FMT_MOF.1(6)	The TOE provides the ability to enable or disable the available TOE services for unauthenticated users (ICMP) through the provided CLI interface. This functionality is limited to the Security Administrator.
FMT_MOF.1(7)	The TOE provides the ability to determine the behavior of the quota control mechanism through the provided CLI interface. These mechanisms include: Controlled connection-oriented resource allocation (FRU_RSA.1(2)), An administrator-specified network identifier, set of administrator-specified network identifiers, administrator-specified period of time. This functionality is limited to the Security Administrator.
FMT_MOF.1(8)	The TOE provides the ability to enable, disable, determine and modify the behavior the configurable number of unsuccessful authentication attempts through the provided CLI interface. This functionality is limited to the Security Administrator. Note: The lockout feature does not apply to a user logging into the TOE locally since it does not make sense to lock a local administrator's account in this fashion
FMT_MSA.1(1)	The TOE provides the ability to manipulate the attributes used to enforce the UNAUTHENTICATED INFORMATION FLOW SFP, and the UNAUTHENTICATED TOE SERVICES SFPs through the provided CLI interface. This functionality is limited to the Security Administrator.
FMT_MSA.1(2)	The TOE provides the ability to manipulate the attributes used to enforce the VPN SFP through the provided CLI interface. This functionality is limited to the Security Administrator.
FMT_MSA.1(3)	The TOE provides the ability to query, modify, or delete the attributes used to enforce the UNAUTHENTICATED INFORMATION FLOW SFP through the

TOE SFRs	How the SFR is Met
	provided CLI interface. This functionality is limited to the Security Administrator.
FMT_MSA.3(1)	The default TOE SFP is restrictive for the VPN SFP implemented within the TOE. Information flows must be administratively configured to be allowed. The TOE only allows the Security Administrator to specify alternate initial values for the attributes used to enforce the SFP.
FMT_MSA.3(2)	The default TOE SFP is restrictive for the UNAUTHENTICATED INFORMATION FLOW SFP implemented within the TOE. Information flows must be administratively configured to be allowed. The TOE only allows the Security Administrator to specify alternate initial values for the attributes used to enforce the SFP.
FMT_MSA.3(3)	The default TOE SFP is restrictive for the UNAUTHENTICATED TOE SERVICES SFP implemented within the TOE. With the exception of ICMP replies, which are allowed by default, information flows must be administratively configured to be allowed. The TOE only allows the Security Administrator to specify alternate initial values for the attributes used to enforce the SFP.
FMT_MTD.1(1)	The ASR provides the ability to query, modify, delete, and clear all information (excluding cryptographic data, audit data, and time data) within the TOE this includes the parameters associated with the BGP, IS-IS, OSPF, PIM, and RIP routing protocols. This is managed through the provided CLI interface by the Security Administrators.
FMT_MTD.1(2)	The ASR provides the ability to modify the all cryptographic security information within the ASR 1000 TOE. This is managed through the provided CLI interface. This functionality is restricted to the Cryptographic Administrator.
FMT_MTD.1(3)	The TOE provides the ability to set the time and date within the TOE. This is managed through the provided CLI interface by the Security Administrator.
FMT_MTD.1(4)	The TOE provides the ability to query, modify, delete, and create the all information associated with information flow configuration (information flow policy rules) within the TOE. All information flow configuration is managed through the provided CLI interface. These actions are restricted to the Security Administrator.
FMT_MTD.2(1)	The TOE maintains counters of the number of "half-open" transport layer connections, as well as, the total connection rate through the TOE. When the TOEs counters exceed the maximum incomplete sessions (max-incomplete) or the maximum high sessions over one minute, the router will reset one old half-open connection for every new connection that exceeds the configured max-incomplete or one-minute high values (Number of new unestablished sessions that will cause the system to start deleting half-open sessions), until the number of half-open sessions drops below the max-incomplete low values (Number of new unestablished sessions that will cause the system to stop deleting half-open sessions). If the total number of connections exceeds the configured maximum number of sessions, the TOE does not accept or create any new connections until the total connections drops below the configured maximum. The functionality applies to both TCP and User Datagram Protocol (UDP) connections over the transport layer. Although UDP is a connectionless protocol, the TOE regards UDP sessions with traffic in only one direction as "half-open,". UDP sessions without return traffic are likely indicative of DoS activity or attempts to connect between two hosts where one of the hosts has become unresponsive.
FMT_MTD.2(2)	The TOE maintains counters of the number of "half-open" TCP connections, as well as, the total connection rate through the TOE. When the TOEs counters exceed the maximum incomplete sessions (max-incomplete) or the maximum high sessions over one minute (one-minute high), the router will reset one old half-open connection for every new connection that exceeds the configured max-incomplete or one-minute high values, until the number of half-open sessions drops below the max-incomplete low values. If the total number of connections

TOE SFRs	How the SFR is Met
	exceeds the configured maximum number of sessions, the TOE does not accept or create any new connections until the total connections drops below the configured maximum. While this functionality is applicable to all connection oriented protocol for which the TOE processes, the TOE only supports TCP connections. For the purpose of this SFR, Connection oriented resources refer to TCP resources.
FMT_MTD.2(3)	The TOE provides the ability to configure the TOEs ability to handle the event that the storage allocated for audit records become used beyond an acceptable percentage of available space. The TOE allows the Security Administrator to configure how to handle the connections after the quota has been met. The TSF takes one of the following actions when the limit is exceeded: • Terminates all active processes or • Overwrites the oldest audit records with the newest records, per administrative configuration
FMT_REV.1	Since TOE does not allow users other than administrators the requirement to revoke users' security attributes is de jure fulfilled. The TOE provides the Security Administrator the ability to revoke the security attributes associated with information flow policies. These permissions can be configured through the TOE provided CLI administrative interface. The following rules are enforced by the TSF with regards to revocations: • changes to the information flow policy ruleset when applied; • disabling of a service available to unauthenticated users; • changes to the set of security associations with peer TOEs; • revocation of the user's role (Security, Crypto, Audit Administrator)
FMT_SMF.1	The TOE provides all the capabilities necessary to securely manage the TOE, the services provided by the TOE, and the information flows through the TOE. The management functionality of the TOE is provided through the TOE CLI. The specific management capabilities available from the TOE are identified in the text of FMT_SMF.1.
FMT_SMR.2	The TOE maintains several administrative roles including, an Audit Administrator, Cryptographic Administrator, Security Administrator. The TOE also supports Authorized IT entities which communicate with the TOE. Users are able to be associated with roles by authenticating to the TOE. Additionally, the following conditions are enforced by the TOE, • all roles shall be able to administer the TOE locally; • all roles shall be able to administer the TOE remotely; • all roles are distinct; that is, there shall be no overlap of operations performed by each role, with the following exceptions: • all administrators can review the audit trail; and • all administrators can invoke the self-tests with the exception of cryptographic self-tests which can only be invoked by the Cryptographic Administrator. Note that the cryptographic and non-cryptographic self tests are two distinct sets. The cryptographic includes the power-on self tests needed to pass FIPS validations, and the non-cryptographic self-tests includes the image integrity check.
FPT_FLS.1	Whenever any failure occurs within the TOE that results in the TOE ceasing operation, the TOE securely disables its interfaces to prevent the unintentional flow of any information to or from the TOE. This functionally prevents any failure from causing an unauthorized information flow. There are no failures that circumvent this protection. Note that the Finite State Module that was created and examanined as part of the FIPS validation covers all states and shows that they fail appropriately (closed).

TOE SFRs	How the SFR is Met
FPT_HA_(EXT).1	For TOE configurations that include dual ESPs and RPs, whenever a failure occurs within the TSF, the TOE automatically switches over to a hot standby to continue operations. This includes any hardware failure within an ESP or RP and any software failure.
FPT_ITA.1	The TOE supports multiple routing protocols including, BGP, PIM, IS-IS, OSPF, RIP, and MPLS. The TOE meets the availability metrics defined in each of the protocols. As long as the TOE is connected to a network, the TOE meets the availability metrics specific in each protocol. For each protocol, the availability timers defined within the protocol are not exceeded. The specific timers for each protocol are identified in the text of the SFR. The TOE fully implements each of the supported routing protocols. Each of the suggested protocol metrics are enforced by the TOE. The availability parameters for BGP, IS-IS, OSPF, PIM, and RIP are configurable by an administrator.
FPT_ITC.1	The TOE provides the ability to create an IPSec tunnel between external IT devices (VPN Peers). The communications between the TOE and these IT devices are protected from authorized disclosure by the encryption capabilities of IPSec. Any traffic sent through or from the TOE through an IPSec connection is protected. The traffic is protected whether the TOE itself or the VPN peer creates initiates the IPSec connection.
FPT_ITI.1	The TOE provides the ability to create an IPSec tunnel between external IT devices (e.g., peer routers). The communications between the TOE and these IT devices are protected from authorized modification by the data integrity capabilities of IPSec. If the data transmitted through an IPSec tunnel is identified as modified, the data is discarded and not used by the TOE. Any traffic sent through or from the TOE through an IPSec connection is protected. The traffic is protected whether the TOE itself or the VPN peer creates initiates the IPSec connection.
FPT_PRO_(EXT).1	The TOE implements several protocols in support of the TOE routing capabilities. These protocols include PIM, IS-IS, BGP, OSPF, RIP, and MPLS.
FPT_RCV.1	Whenever the TOE experiences a fault from which it cannot automatically recover, the TOE enters a safe state that allows the administrators to return the TOE to an operational state.
FPT_RCV.2	The TOE provides the ability to automatically recover from a single fault in either the active RP/ESP or standby RP/ESP for TOE configurations that include dual RPs or ESPs. This includes the ability to handle both hardware and software failures. The TOE also provides the ability to automatically recover from a RNG stuck condition in which the implemented RNG creates two consecutive identical random sequences. All other faults result in the TOE ceasing interface transmissions until the fault is addressed by user interaction.
FPT_RPL.1	Attempted replay packets against administrative traffic are detected and then discarded. The SSHv2 protocol used for management of the TOE contains a mechanism to identify traffic that is corrupted (either because of a replay attack or because of a random occurrence). Whenever a packet with a problem is received by the TOE, the TOE discards the packet. Attempted replay packets against IPSec traffic are detected and then discarded. The IPSec protocol used for data plane traffic transmission contains a mechanism to identify traffic that is corrupted (either because of a replay attack or because of a random occurrence). Whenever a packet with a problem is received by the TOE, the TOE discards the packet.
FPT_STM.1	The TOE internally maintains the date and time. This date and time is used as the time stamp that is applied to TOE generated audit records.
FPT_TDC.1	The TOE fully supports the following protocols, PIM, IS-IS, BGP, OSPF, RIP, and MPLS. When data associated with each of these protocols is received by the TOE, the TOE is able to interpret the data.

TOE SFRs	How the SFR is Met
FPT_TST.1(1)	The TOE performs cryptographic self-tests consistent with the requirements of FIPS 140-2. All cryptographic algorithms implemented within the TOE are subject to the applicable self-tests. These tests are performed at start up, by the cryptographic administrator on demand, and periodically, as applicable (whenever an RNG is used or an asymmetric key pair is created) per the FIPS 140-2 requirements. Please see the Security Policy associated with FIPS 140-2 certificate # 1390 for additional information regarding the FIPS self-test implemented by the TOE. This includes the capability to verify the integrity of TSF data and executable code related to cryptography. Further details regarding the FIPS validation can be found in Certificate #1390, Cisco ASR1002 Series Router, Cisco ASR1004 Series Router, Cisco ASR1006 Series Router.
FPT_TST.1(2)	The TOE performs cryptographic self-tests consistent with the requirements of FIPS 140-2. All cryptographic algorithms implemented within the TOE are subject to the applicable self-tests. Please see the Security Policy associated with FIPS 140-2 certificate # 1390 for additional information regarding the FIPS self-test implemented by the TOE. This includes the capability to verify the integrity of TSF data and executable code related to cryptography and key generation. Further details regarding the FIPS validation can be found in Certificate #1390, Cisco ASR1002 Series Router, Cisco ASR1004 Series Router, Cisco ASR1006 Series Router.
FPT_TST_(EXT).1	The TOE performs an integrity verification of the executable code within the TOE. The integrity check is a SHA-1 hash as allowed by the FIPS 140-2 standard. These tests are performed during initial startup of the TOE. The integrity self-tests was verified as part of the FIPS 140-2 validation. Further details regarding the FIPS validation can be found in Certificate #1390, Cisco ASR1002 Series Router, Cisco ASR1004 Series Router, Cisco ASR1006 Series Router. This functionality can be executed by all administrators by issuing a reload of the TOE.
FRU_RSA.1(1)	The TOE provides the ability to limit transport-layer quotas. This functionality allows the TOE to limit the transport-layer traffic available to a specific set of IP addresses over an administrator specified period of time. Specifically, when the configured number of maximum sessions is exceeded, new connections are not accepted /created. When the configured number of maximum incomplete sessions is exceeded, the oldest half-opened sessions are deleted. If there are no half-open sessions, new connections are not accepted/created. And when the configured number of maximum high/low sessions over an administrator configured period of time is exceeded, the oldest half-opened sessions are deleted. If there are no half-open sessions, new connections are not accepted/created.
FRU_RSA.1(2)	The TOE provides the ability to limit controlled connection-oriented resources. This functionality allows the TOE to limit users associated with an administrator-specified network identifier and a set of administrator-specified network identifiers can use over an administrator-specified period of time. Specifically, when the configured number of maximum sessions is exceeded, new connections are not accepted /created. When the configured number of maximum incomplete sessions is exceeded, the oldest half-opened sessions are deleted. If there are no half-open sessions, new connections are not accepted/created. And when the configured number of maximum high/low sessions over an administrator configured period of time is exceeded, the oldest half-opened sessions are deleted. If there are no half-open sessions, new connections are not accepted/created.
FTA_SSL.1	The TOE locks a local administrative session after a configurable amount of idle time. The amount of idle time required to lock a local administrative session is only able to be configured by the Security Administrator. When the local session

TOE SFRs	How the SFR is Met
	is locked, the screen is flushed and the no further activity is allowed until the administrator re-authenticates to the TOE.
FTA_SSL.2	The TOE allows the administrators connected to the TOE locally to lock their administrative sessions on demand. When the local session is locked, the screen is flushed and the no further activity is allowed until the administrator re-authenticates to the TOE.
FTA_SSL.3	The TOE terminates a remote administrative session after a configurable amount of idle time. The amount of idle time required to terminate a remote administrative session is only able to be configured by the Security Administrator.
FTA_TAB.1	The TOE displays a Security Administrator specified banner on the CLI management interface prior to allowing any administrative access to the TOE.
FTA_TSE.1	The TOE provides the ability to deny administrative access to the TOE management CLI based on the location (IP address) of the requesting administrator, and the time and day of the connection request.
FTP_ITC.1(1)	The TOE provides the ability to communicate with remote IT entities (VPN Peers) through an IPSec encryption secured tunnel that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from disclosure. Either the TSF or the authorized IT entity (VPN Peers) to initiate the encrypted session. This prevents the data sent between the TOE and the remote IT entity (VPN Peers) from disclosure. The types of information that are sent through the IPSec encrypted tunnel include authentication related data and network control data (PIM, IS-IS, BGP, OSPF, RIP, and MPLS.) Assured identification of VPN peers is achieved via the use of the IKE protocol as specified in FCS_IKE_(EXT).1. Certificate revocation for VPN peers is supported via CRL (RFC 3280) and OCSP (RFC 2560). When a trustpoint (a CA) is configured, both CRL checks and OCSP revocation checks may be configured to provide certificate revocation.
FTP_ITC.1(2)	The TOE provides the ability to communicate with remote IT entities (VPN Peers) through an IPSec encryption secured tunnel. Either the TSF or the authorized IT entity (VPN Peers) to initiate the encrypted session This prevents the data sent between the TOE and the remote IT entity (VPN Peers) from unauthorized/undetected modification. The types of information that are sent through the IPSec encrypted tunnel include authentication related data and network control data (PIM, IS-IS, BGP, OSPF, RIP, and MPLS).
FTP_TRP.1(1)	All remote administrative communications take place over a secure encrypted SSHv2 session. The SSHv2 session is encrypted using AES encryption. The remote users are able to initiate SSHv2 communications with the TOE. The types of information that are sent through the SSHv2 encrypted tunnel include authentication related data and administrative actions.
FTP_TRP.1(2)	All remote administrative communications take place over secure integrity-verified SSHv2 session. The SSHv2 session is encrypted using AES encryption. The remote users are able to initiate SSHv2 communications with the TOE. The types of information that are sent through the SSHv2 encrypted tunnel include authentication related data and administrative actions.

6.2 TOE Bypass and Interference/Logical Tampering Protection Measures

The ASR consists of a hardware platform in which all operations in the TOE scope are protected from interference and tampering by untrusted subjects. All administration and configuration operations are performed within the physical boundary of the TOE. Also, all TSP enforcement functions must be invoked and succeed prior to functions within the TSC proceeding.

The TOE has been designed so that all locally maintained TSF data can only be manipulated via the secured management interface, a CLI interface. There are no undocumented interfaces for managing the product.

All sub-components included in the TOE (RPs and ESPs) rely on the main ASR 1000 chassis for power, memory management, and access control. In order to access any portion of the TOE, the Identification & Authentication mechanisms of the ASR must be invoked and succeed.

No processes outside of the ASR are allowed direct access to any TOE memory. The TOE only accepts traffic through legitimate TOE interfaces. None of these interfaces provide any access to internal TOE resources.

The ASR provides a private resource space for each VLAN to operate within. Each VLAN has its own forwarding plane resources that other VLANs within the same ASR switch TOE component are not able to affect.

The ASR provides a private resource space for each VRF to operate within. Each VRF has its own resources that other VRFs within the same ASR switch TOE component are not able to affect.

Finally, the ASR enforces information flow control policies and applies network traffic security on its interfaces before traffic passes into or out of the TOE. The TOE controls every ingress and egress traffic flow. Policies are applied to each traffic flow. Traffic flows characterized as unauthorized are discarded and not permitted to circumvent the TOE.

There are no unmediated traffic flows into or out of the TOE. The information flow policies identified in the SFRs are applied to all traffic received and sent by the ASR. Each communication including data plane communication, control plane communications, and administrative communications are mediated by the TOE. There is no opportunity for unaccounted traffic flows to flow into or out of the TOE.

This design, combined with the fact that only an administrative user with the appropriate role may access the TOE security functions, provides a distinct protected domain for the TOE that is logically protected from interference and is not bypassable.

7 RATIONALE

This section describes the rationale for the Security Objectives and Security Functional Requirements as defined within this Security Target. Additionally, this section describes the rationale for not satisfying all of the dependencies. The table below illustrates the mapping from Security Objectives to Threats and Policies.

7.1 Rationale for TOE Security Objectives

NOTE: The Objectives rationale included in this ST is drawn from the three Medium Robustness Protection Profiles for which conformance is claimed. Similar Objective Rationale or Objective Rationale with the same purpose have been combined where appropriate. The objectives to TOE SFRs mappings have also been updated to reflect the consolidated list of TOE SFRs.

Table 28: Threat/Policies/Objectives/SFRs Mappings/Rationale

Threat/ Policy	Objective	Rationale
T.ADDRESS_MASQUERADE	O.MEDIATE_INFORMATION_FLOW (AKA O.MEDIATE)	O.MEDIATE (FDP_IFC.1(4), FDP_IFF.1(4), FDP_IFC.1(2), FDP_IFF.1(2)) counters this threat by ensuring that all network packets that flow through the TOE are subject to the information flow policies. One of the rules in each of the policies ensures 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	O.ROBUST_ADMIN_GUIDANCE	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 insecure.
	O.ADMIN_ROLE	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. For example, the Audit Administrator could not make a configuration mistake that would impact the directory access control policy. Likewise, a directory manager could only affect policies in the sub-hierarchy they are responsible for, and not other sub-hierarchies or global directory policies.
	O.MEDIATE_INFORMATION_FLOW (AKA O.MEDIATE)	O.MANAGE (FMT_MTD.1(1), FMT_MTD.1(2), FMT_MTD.1(3), FMT_MTD.1(4), FMT_SMF.1) 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

Threat/ Policy	Objective	Rationale
		mistakes that might have been made.
T.ADMIN_ROGUE	O.ADMIN_ROLE	O.ADMIN_ROLE (FMT_SMR.2) mitigates this threat by restricting 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 directory access control policy. On the other hand, if the Security Administrator becomes malicious they could affect the directory access control policy, but the Audit Administrator may be able to detect those actions.
T.AUDIT_COMPROMISE	O.AUDIT_PROTECTION	O.AUDIT_PROTECTION (FAU_SAR.2, FAU_STG.1, FAU_STG.3, FAU_STG_(EXT).4, FMT_MOF.1(2), FMT_MOF.1(4), FMT_MOF.1(5), FMT_MOF.1(6), FMT_MOF.1(8)) 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, and of notifying an administrator if the audit trail is approaching its capacity. In addition, the TOE has the capability to restore audit data corrupted by the attacker.
	O.RESIDUAL_INFORMATION	O.RESIDUAL_INFORMATION (FDP.RIP.2) 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	O.SELF_PROTECTION (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.
T.CRYPTO_COMPROMISE	O.RESIDUAL_INFORMATION	O.RESIDUAL_INFORMATION (FCS_CKM.4) mitigates the possibility of malicious users or processes from gaining inappropriate access to cryptographic data, including keys. This objective ensures that the cryptographic data does not reside in a resource that has been used by the cryptographic module and then reallocated to another process. This objective is necessary to mitigate this threat by ensuring no TSF data remain in resources allocated to a user. 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	O.SELF_PROTECTION (ADV_ARC) 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

Threat/ Policy	Objective	Rationale
		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. If the TSF could not maintain and control its domain of execution, it could not be trusted to control access to the resources under its control, which includes the cryptographic data and executable code.
T.MASQUERADE	O.ROBUST_TOE_ACCESS	O.ROBUST_TOE_ACCESS (FIA_AFL.1, FIA_ATD.1(1), FIA_ATD.1(2), FIA_UID.2, FIA_UAU.1, FIA_UAU.2, FIA_UAU_(EXT).5, FTA_TSE.1, AVA_VAN.3) 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.
	O.TRUSTED_PATH	O.TRUSTED_PATH (FTP_ITC.1(1), FTP_ITC.1(2)) 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 compromised or disclosed (e.g., encrypted). The protection offered by this objective is limited to TSF data and security attributes.
	O.USER_GUIDANCE	O.USER_GUIDANCE (AGD_USR.1) instructs users on the proper use of logging into the proxy servers, which plays a role in mitigating the threat of an attacker attempting to masquerade as an authorized user by providing users with the proper guidance on how to securely login to the proxies.
T.FLAWED_DESIGN	O.CHANGE_MANAGEMENT	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	O.SOUND_DESIGN (ADV_FSP_.4, ADV_TDS.3, ADV_INT_.1,) 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. The design of the TOE can be better understood, which increases the chances that design errors will be discovered.
	O.VULNERABILITY_ANALYSIS_TEST	O.VULNERABILITY_ANALYSIS_TEST (AVA_VAN.3) ensures that 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.
T.FLAWED_IMPLEMENTATIO	O.CHANGE_MANAGEMENT	O.CHANGE_MANAGEMENT (ALC_CMC.4, ALC_CMS.4, ALC_DVS.1, ALC_FLR.2, ALC_LCD.1,) This objective plays a role

Threat/ Policy	Objective	Rationale
N		in mitigating this threat in the same way that the poor 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.
	O.SOUND_IMPLEMENTATION	In addition to documenting the design so that implementers have a thorough understanding of the design, O.SOUND_IMPLEMENTATION ADV_TDS.3, ADV_INT.1, (ADV_IMP.2, 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.
	O.THOROUGH_FUNCTIONAL_TESTING	Although the previous three objectives help minimize the introduction of errors into the implementation, O.THOROUGH_FUNCTIONAL_TESTING (ATE_COV.2, ATE_FUN.1, ATE_DPT.1, 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	O.VULNERABILITY_ANALYSIS_TEST (AVA_VAN.3) helps reduce errors in the implementation that may not be discovered during functional testing. Ambiguous design documentation, and the fact that exhaustive testing of the external interfaces is not required may leave bugs in the implementation undiscovered in functional testing. Having an independent party perform a vulnerability analysis and conduct testing outside the scope of functional testing increases the likelihood of finding errors.
T.POOR_TEST	O.CORRECT_TSF_OPERATION	While these testing activities are a necessary activity for successful completion of an evaluation, this testing activity does not address the concern that the TOE continues to operate correctly and enforce its security policies once it has been fielded. Some level of testing must be available to end users to ensure the TOE's security mechanisms continue to operate correctly once the TOE is fielded O.CORRECT_TSF_OPERATION (FPT_TST_(EXT).1, Crypto Self-Test (FPT_TST.1(1), and) Key Generation Self-Test (FPT_TST.1(2)) 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	Design analysis determines that TOE's documented design satisfies the security functional requirements. In order to ensure the TOE's design is correctly realized in its implementation, the appropriate level of functional testing of the TOE's security mechanisms must be performed during the evaluation of the TOE. O.THOROUGH_FUNCTIONAL_TESTING (ATE_FUN.1, ATE_COV.2, ATE_DPT.1, 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

Threat/ Policy	Objective	Rationale
		unintended ways to circumvent the TOE's security policies.
	O.VULNERABILITY_ANALYSIS_TEST	O.VULNERABILITY_ANALYSIS_TEST (AVA_VAN.3) addresses this concern by requiring a vulnerability analysis be performed in conjunction with testing that goes beyond functional testing. This objective provides a measure of confidence that the TOE does not contain security flaws that may not be identified through functional testing.
T.REPLAY	O.REPLAY_DETECTION	O.REPLAY_DETECTION (FPT_RPL.1) prevents a user from replaying TSF data and security attributes (e.g., TSF data or security attributes transmitted between a remote administrator, the authentication server, an authorized IT entity and the TOE) that could leave the TOE in a configuration that the administrative staff did not intend (e.g., an administrator modifies the auditable events to be recorded and a user captures that traffic. At a later date the administrator determines that the new set of auditable events is not sufficient and again modifies the events to be audited. The user then replays the earlier audit event configuration.) O.REPLAY_DETECTION (FPT_RPL.1) prevents a user from replaying authentication data. Prevention of replay of authentication data will counter the threat that a user will be able to record an authentication session between a trusted entity (administrative user or trusted IT entity) and then replay it to gain access to the TOE, and counter the ability of a user to act as another user.
	O.ROBUST_TOE_ACCESS	O.ROBUST_TOE_ACCESS (FIA_UAU_(EXT).5) contributes to countering this threat by requiring the TOE have the capability to invoke a single-use authentication mechanism. A single-use authentication mechanism ensures that once authentication data has been presented to authenticate a user, that authentication data cannot be used again, therefore a user could not capture authentication and reuse it at a later time.
T.RESIDUAL_DATA	O.RESIDUAL_INFORMATION	O.RESIDUAL_INFORMATION (FDP_RIP.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 sent in response to a request will not have residual data from another packet (potentially from another user) due to the padding of a packet. The TSF data will be zeroized once the resources are released by a user/process. This also ensures that memory that is made available to proxy users will not contain residual data.
T.RESOURCE_EXHAUSTION	O.RESOURCE_SHARING	O.RESOURCE_SHARING (FRU_RSA.1(1), FRU_RSA.1(2), FMT_MTD.2(1), FMT_MTD.2(2), FMT_MTD.2(3), FMT_MOF.1(7), FMT_SMF.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	O.TRUSTED_PATH	It is possible for an entity other than the TOE (a subject on the TOE, or another IT entity) to provide an environment that may lead a user to mistakenly believe they are interacting with the TOE thereby

Threat/ Policy	Objective	Rationale
		fooling the user into divulging identification and authentication information. O.TRUSTED_PATH (FTP_ITC.1(1), FTP_ITC.1(2), FTP_TRP.1(1), FTP_TRP.1(2)) 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	O.DISPLAY_BANNER	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_INFORMATION_FLOW (AKA O.MANAGE)	O.MANAGE (FMT_MTD.1(1)-(4), FMT_MSA.1(1), FMT_MSA.1(2), FMT_MSA.1(3), FMT_MSA.3(1) - (3), FMT_MOF.1(1)-(8)) 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. This objective provides the capability to restrict access to TSF to those that are authorized to use the functions. Satisfaction of this objective (and its associated requirements) prevents unauthorized access to TSF functions and data through the administrative mechanisms.
	O.RESIDUAL_INFORMATION	O.RESIDUAL_INFORMATION (FDP_RIP.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	O.SELF_PROTECTION (ADV_ARC.1, FTP_TRP.1, FTP_ITC.1) requires that the TSF be able to protect itself from tampering and that the security mechanisms in the TSF cannot be bypassed. Without this objective, there could be no assurance that users could not view or modify TSF data or TSF executables. This objective 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. (ADV_ARC.1) requires that the TSF be able to protect itself from tampering and that the security mechanisms in the TSF cannot be bypassed. Without this objective, there could be no assurance that users could not view or modify TSF data or TSF executables.
	O.TRUSTED_PATH	O.TRUSTED_PATH (FTP_TRP.1(1), FTP_TRP.1(2), FTP_ITC.1(1), FTP_ITC.1(2)) 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, including authentication data and all data sent or received by trusted IT entities (a relying party's user data is not protected; only the authentication

Threat/ Policy	Objective	Rationale
		portion of the session is protected), and security attributes (e.g., the data communication between peer TOEs via a VPN is protected by the VPN policy stated in FDP_IFC.1(1) and FDP_IFF.1(1) and FTP_ITC does not apply to VPN communications) and (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	O.ROBUST_TOE_ACCESS	O.ROBUST_TOE_ACCESS (FTA_SSL.1, FTA_SSL.2, FTA_SSL.3) 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	O.MEDIATE	O.MEDIATE (FDP_IFF.1(4), FDP_IFC.1(4), FMT_REV.1, ADV_ARC.1, FDP_IFC.1(2), FDP_IFF.1(2)) works to mitigate this threat by ensuring that all network packets that flow through the TOE are subject to the information flow policies. One of the rules ensures that the network identifier in a 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 purported to originate from a network 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. Another rule provides further granularity of access control by enabling the administrator to control the source and destination address, destination port, protocol, and application level commands. By implementing this level of access control an attacker would not be allowed access to other hosts and applications residing on the protected network. Additionally, the TOE maintains "state" information of all approved connections. This function ensures that each packet arriving at a TOE interface purporting to be part of an approved connection through or to the TOE, is checked against a current and valid list of connection parameters (e.g. for a TCP/IP connection; source and destination address, ports, SYN and ACK numbers, flags, etc.) prior to allowing the packet through or to the TOE. The VPN policy ensures that user data being sent between PEER TOEs is encrypted if there is a rule (specified by the Security Administrator) that states data is to be encrypted between those two hosts. The VPN policy allows the administrator to specify for each originating host (identified by IP address), which destination addresses must be accessed through a VPN (using ESP tunnel mode) and which destination addresses may be accessed without VPN encryption. If a potential security violation has been detected, the TOE displays a message that identifies the potential security violation to all administrative consoles. The consoles include the local TOE console and any active remote administrative sessions. If an

Threat/ Policy	Objective	Rationale
		administrator is not currently accessing the TOE, the message is stored and immediately displayed the next time an administrator accesses the TOE. The authenticated TOE policy allows the administrator to specify each originating host (identified by IP address), which destination addresses must be access through a router and which destination addresses may be accessed without encryption. If a potential security violation has been detected, the TOE displays a message that identifies the potential security violation to all administrator consoles. The consoles include the local TOE console and any active remote administrative sessions. If an administrator is not currently accessing the TOE, the message is stored and immediately displayed the next time an administrator accesses the TOE. The TOE requires successful authentication through a protected communication path (with account lock-out capability) to the TOE prior to gaining access to certain services on or mediated by the TOE. By implementing strong authentication to gain access to these services, an attacker's opportunity to successfully conduct a man-in-the-middle and/or password guessing attack is greatly reduced. Lastly, the TSF must ensure that all configured enforcement functions (authentication, access control rules, etc.) must be invoked prior to allowing a user to gain access to TOE or TOE mediated services. The TOE restricts the ability to modify the security attributes associated with access control rules, access to authenticated and unauthenticated services, etc., to the Security Administrator. This feature ensures that no other user can modify the information flow policy to bypass the intended TOE security policy. 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.USER_GUIDANCE	O.USER_GUIDANCE (AGD_USR.1) mitigates this threat by providing the user the information necessary to use the security mechanisms that control access to user data in a secure manner. For instance, the method by which the discretionary access control mechanism is configured, and how to apply it to the data the user owns, is described in the user guidance. If this information were not available to the user, the information may be left unprotected, or the user may mis-configure the controls and unintentionally allow unauthorized access to their data.
T.UNAUTHORIZED_PEER	O.PEER_AUTHENTICATION	O.PEER_AUTHENTICATION (FCS_IKE_(EXT).1) mitigates this threat by requiring that the TOE implement the Internet Key Exchange protocol, as specified in RFC2409, to establish a secure, authenticated channel between the TOE and another remote VPN endpoint before establishing a security association with that remote endpoint or another remote router before establishing a security association with that router. O.PEER_AUTHENTICATION (FCS_FDOI_(EXT).1 mitigates this threat by requiring that the TOE implement the GDOI protocol, as specified in RFC 3547, as an extension to RFC2409. This protocol is used to establish security associations between groups of IPSec users.

Threat/ Policy	Objective	Rationale
T.UNIDENTIFIED_ACTIONS	O.AUDIT_REVIEW	O.AUDIT_REVIEW (FAU_SAA.1, FAU_ARP.1, FAU_SAR.1, FAU_SAR.3, FAU_ARP_ACK.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. By configuring these auditable events, the TOE monitors the occurrences of these events (e.g. set number of authentication failures, set number of information policy flow failures, self-test failures, etc.) and immediately notifies all TOE administrators once an event has occurred or a set threshold has been met. If a potential security violation has been detected, the TOE displays a message that identifies the potential security violation to all administrative consoles. The consoles include the local TOE console and any active remote administrative sessions. If an administrator is not currently logged into the TOE, the message is stored and immediately displayed the next time an administrator accesses the TOE. This message is displayed to all administrative roles and will remain on the screen for each administrative role until each administrative role acknowledges the message. In addition to displaying the potential security violation, the message must contain all audit records that generated the potential security violation. By enforcing the message content and display, this objective provides assurance that a TOE administrator will be notified of a potential security violation. The TOE can also be configured to generate an audible alarm, which may alert administrators who are not sitting at their administrative workstation or console. The TOE also requires an Audit Administrative role. This role is restricted to Audit record review and the deletion of the audit trail for maintenance purposes. A search and sort capability provides an efficient mechanism for the Audit Administrator to view pertinent audit information. For analyzing the audit trail, the TOE requires an Auditor role. This role is restricted to Audit record review and the deletion of the audit trail for maintenance purposes. A search and sort capability provides an efficient mechanism for the Audit Administrator to view pertinent audit information. In addition to the local Auditor role, the TOE also has the capability to export the audit information to an external audit analysis tool (such as an intrusion detection system) for more detailed or composite audit analysis. The TOE's audit analysis mechanism must consist of a minimum set of configurable audit events that could indicate a potential security violation. Thresholds for these events must be configurable by an appropriate administrative role. By configuring these auditable events, the TOE monitors the occurrences of these events (e.g. set number of authentication failures, set number directory access failures, self-test)
T.UNKNOWN_STATE	O.MAINT_MODE	O.ROBUST_ADMIN_GUIDANCE (AGD_OPE.1, AGD_PRE.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).

Threat/ Policy	Objective	Rationale
		O.MAINT_MODE (FPT_RCV.1, FPT_RCV.2) helps to mitigate this threat by ensuring that the TOE does not continue to operate in an insecure state when a hardware or software failure occurs. After a failure, the TOE enters a state that disallows operations and requires an administrator to follow documented procedures to return the TOE to a secure state.
	O.SOUND_DESIGN	O.CORRECT_TSF_OPERATION (FPT_TST_(EXT).1, Crypto Self-Test (FPT_TST.1(1), and) Key Generation Self-Test (FPT_TST.1(2)), counters this threat by ensuring that the TSF runs a suite of tests to successfully demonstrates the correct operation of the TSF's underlying abstract machine (hardware and software), the TSF, and the TSF's cryptographic components at initial startup of the TOE. In addition to ensuring that the TOE's security state can be verified, the Security Administrator can verify the integrity of the TSF's data and stored code as well as the TSF's cryptographic data and stored code. O.SOUND_DESIGN (ADV_ARC.1) works to mitigate this threat by requiring that the TOE developers provide accurate and complete design documentation of the security mechanisms in the TOE, including a security model. By providing this documentation, the possible secure states of the TOE are described, thus enabling the administrator to return the TOE to one of these states during the recovery process.
	O.ROBUST_ADMIN_GUIDANCE	O.MAINT_MODE (FPT_RCV.1, FPT_RCV.2) helps to mitigate this threat by ensuring that the TOE does not continue to operate in an insecure state when a hardware or software failure occurs. After a failure, the TOE enters a state that disallows traffic flow and requires an administrator to follow documented procedures to return the TOE to a secure state. This objective provides administrative guidance for the secure start-up of the TOE and 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 manor. 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).
	O.CORRECT_ TSF_OPERATION	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. This objective counters this threat by ensuring that the TSF runs a suite of tests to successfully demonstrate the correct operation of the TSF (hardware and software) and the TSF's cryptographic components at initial startup of the TOE. In addition to ensuring that the TOE's security state can be verified, an administrator can verify the integrity of the TSF's data and stored code and the TSF's cryptographic data and stored code using the TOE-provided cryptographic mechanisms.
T.EAVESDROP	O.CRYPTOGRAPHIC_FUNCTIONS	O.CRYPTOGRAPHIC_FUNCTIONS (FCS_CKM.1(1), FCS_CKM.1(2), FCS_CKM.2, FCS_CKM_(EXT).2, FCS_CKM.4,

Threat/ Policy	Objective	Rationale
		FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3), FCS_COP.1(4), FCS_COP_(EXT).1 mitigates this threat by providing for the use of cryptographic functions to detect when information has been modified.
	O.PROTECT_IN_TRANSIT	O.PROTECT_IN_TRANSIT (FPT_ITA.1, FPT_ITC.1, FPT_ITI.1, FTP_TRP.1(1), FTP_TRP.1(2), FTP_ITC.1(1), FTP_ITC.1(2) satisfies this threat by ensuring protection of the communication between the TOE and trusted IT entities while transmitting data.
T.NORECOVERY	O.HA	O.HA (FPT_HA_(EXT).1) counters this threat by ensuring that any single failure within the TSF does not prevent TSF performance.
P.ACCESS_BANNER	O.DISPLAY_BANNER	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. This is required to be displayed before an interactive administrative session.
P.ACCOUNTABILITY	O.AUDIT_GENERATION	O.AUDIT_GENERATION (FAU_GEN.1, FAU_GEN.2, FIA_USB.1, FAU_SEL.1) 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.). Attributes used in the audit record generation process are also required to be bound to the subject, ensuring users are held accountable. O.AUDIT_GENERATION (FAU_GEN.1, FAU_GEN.2, FIA_USB.1, FAU_SEL.1) addresses this policy by providing an audit mechanism to record the actions of a specific user, and the capability for an administrator to "pre-select" audit events based on the user ID. The audit event selection function is configurable during run-time to ensure the TOE is able to capture security-relevant events given changes in threat conditions.
	O.ROBUST_TOE_ACCESS	O.ROBUST_TOE_ACCESS (FIA_UID.2, FIA_UAU_(EXT).5, FIA_UAU.2) 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).
	O.TIME_STAMPS	O.TIME_STAMPS (FPT_STM.1, FMT_MTD.1(3)) plays a role in supporting this policy by requiring the TOE to provide a reliable time stamp (configured locally by the Security Administrator). 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.
P.ADMIN_ACCESS	O.ADMIN_ROLE	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

Threat/ Policy	Objective	Rationale
		local administrator (e.g., setting the ruleset).
	O.TRUSTED_PATH	O.TRUSTED_PATH (FTP_TRP.1(1), FTP_TRP.1(2), FTP_ITC.1(1), FTP_ITC.1(2)) 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) must adhere to the same requirements as the remote administrator. Additionally, all trusted IT entities (e.g., trusted peer directories, intrusion detection systems) connect through a protected channel, thus avoiding disclosure and spoofing problems.
P.CRYPTOGRAPHY	O.CRYPTOGRAPHIC_FUNCTIONS	O.CRYPTOGRAPHIC_FUNCTIONS (FCS_CKM.1(1), FCS_CKM.1(2), FCS_CKM.2, FCS_CKM.4, FCS_CKM_(EXT).2, FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3), FCS_COP.1(4), FCS_COP.1(5)) FCS_COP_(EXT).1 implements this policy, requiring a combination of FIPS-validation and non-FIPS-validated cryptographic mechanisms that are used to provide encryption/decryption services, and digital signature functions. Functions include symmetric encryption and decryption, digital signatures, and key generation and establishment functions.
	O.RESIDUAL_INFORMATION	O.RESIDUAL_INFORMATION (FDP_RIP.2, FCS_CKM.4) satisfies this policy by ensuring that cryptographic data are cleared from resources that are shared between users. Keys must be zeroized according to FIPS 140-2.
P.CRYPTOGRAPHIC_FUNCTIONS	O.CRYPTOGRAPHIC_FUNCTIONS	O.CRYPTOGRAPHIC_FUNCTIONS (FCS_BCM_(EXT)).1 implements this policy, requiring a combination of FIPS-validation and non-FIPS-validated cryptographic mechanisms that are used to provide encryption/decryption services, as well as digital signature functions. Functions include symmetric encryption and decryption, digital signatures, as well as key generation and establishment functions.
P.CRYPTOGRAPHY_VALIDATED	O.CRYPTOGRAPHY_VALIDATED	O.CRYPTOGRAPHY_VALIDATED (FCS_BCM_(EXT)) satisfies this policy by requiring the TOE to implement NIST FIPS validated cryptographic services. These services will provide confidentiality and integrity protection of TSF data while in transit to remote parts of the TOE.
	O.RESIDUAL_INFORMATION	O.RESIDUAL_INFORMATION (FDP_RIP.2, FCS_CKM.4) satisfies this policy by ensuring that cryptographic data are cleared from resources that are shared between users. Keys must be zeroized according to FIPS 140-2.
P.VULNERABILITY_ANALYSIS_TEST	O.VULNERABILITY_ANALYSIS_TEST	O.VULNERABILITY_ANALYSIS_TEST (AVA_VAN.3) satisfies this policy by ensuring that an independent analysis is performed on the TOE and penetration testing based on that analysis is performed. Having an independent party perform the analysis helps ensure objectivity and eliminates preconceived notions of the TOE's design and implementation that may otherwise affect the thoroughness of the analysis. The level of analysis and testing requires that an attacker with a moderate attack potential cannot compromise the TOE's ability to enforce its security policies.
P.COMPATIBILITY	O.PROTOCOLS	O.PROTOCOLS (FPT_FLS.1, FPT_PRO_(EXT).1, FPT_TDC.1) satisfies this policy by requiring that standardized protocols are implemented in the TOE to ensure interoperatability among peer TOEs therefore not compromising the secure state of the router.
P.INTEGRITY	O.INTEGRITY	O.INTEGRITY (FDP_IFC.1(1), FDP_IFF.1(1)) satisfies this policy

Threat/ Policy	Objective	Rationale
		by ensuring that all IPSec encrypted data received from a peer TOE is properly decrypted and authentication verified.

7.2 Rationale for the Security Objectives for the Environment

Table 29: Assumptions/Objectives Mappings/Rationale

Assumption	Environmental Objective Addressing the Assumption	Rationale
A.AVAILABILITY	OE.AVAILABILITY	Network resources shall be available to allow clients to satisfy mission requirements and to transmit information.
A.NO_GENERAL_PURPOSE	OE.NO_GENERAL_PURPOSE	The Router must not include any general-purpose commuting or storage capabilities. This will protect the TSF data from malicious processes.
A.PHYSICAL	OE.PHYSICAL	The TOE, the TSF data, and protected user data is assumed to be protected from physical attack (e.g., theft, modification, destruction, or eavesdropping). Physical attack could include unauthorized intruders into the TOE environment, but it does not include physical destructive actions that might be taken by an individual that is authorized to access the TOE environment.
A.NO_TOE_BYPASS	OE.NO_TOE_BYPASS	The Router must be placed in a position on the network so that information cannot flow between external and internal networks located in different enclaves without passing through the TOE.

7.3 Rationale for SFRs-SARs/TOE Objectives

This section provides rationale for the Security Functional Requirements/Security Assurance Requirements demonstrating that the Security Functional Requirements/Security Assurance Requirements are suitable to address the security objectives. This section provides mapping rational between each of the identified Security Functional Requirements/Security Assurance Requirements and the mapped TOE security objectives.

Table 30: Objective to SFR Mappings

Objective	Req.	Rationale
O.ADMIN_ROLE	FMT_SMR.2	FMT_SMR.2 requires that three roles exist for administrative actions: the Security Administrator, who is responsible for configuring most security-relevant parameters on the TOE; the Cryptographic Administrator, who is responsible for managing the security data that is critical to the cryptographic operations; and the Audit Administrator, who is responsible for reading and deleting the audit trail. The TSF is able to associate a human user with one or more roles and these roles isolate administrative functions in that the functions of these roles do not overlap. It is true that the design of some systems could enable a rogue security administrator to manipulate cryptographic data by, for instance, writing directly to kernel memory. While this scenario is a security concern, this

		objective does not counter that aspect of T.ADMIN_ROGUE. If a security administrator were to perform such an action, the auditing requirements (along with the audit trail protection requirements) afford some measure of detectability of the rogue administrator's actions.
O.AUDIT_GENERATION	FAU_GEN.1	FAU_GEN.1-NIAP-0429 defines the set of events that the TOE must be capable of recording. This requirement ensures that an administrator has the ability to audit any security relevant event that takes place in the TOE. This requirement also defines the information that must be contained in the audit record for each auditable event. There is a minimum of information that must be present in every audit record and this requirement defines that, and the additional information that must be recorded for each auditable event. This requirement also places a requirement on the level of detail that is recorded on any additional security functional requirements an ST author adds to this PP.
	FAU_GEN.2	FAU_GEN.2-NIAP-0410 ensures that the audit records associate a user identity with the auditable event. Although the FIA_ATD.1 requirements mandate that a "userid" be used to represent a user identity, the TOE developer is able to associate different types of user-ids with different users in order to meet this objective.
	FIA_USB.1	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 authenticated 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.
	FAU_SEL.1	FAU_SEL.1-NIAP-0407 allows the selected administrator(s) to configure which auditable events will be recorded in the audit trail. This provides the administrator with the flexibility in recording only those events that are deemed necessary by site policy, thus reducing the amount of resources consumed by the audit mechanism and providing the ability to focus on the actions of an individual user. In addition, the requirement has been refined to require that the audit event selection function is configurable during run-time to ensure the TOE is able to capture security-relevant events given changes in threat conditions.
O.AUDIT_PROTECTION	FMT_MOF.1(3) FAU_SAR.2 FAU_STG.1 FAU_STG.3 FAU_STG_(EXT).4 FMT_SMF.1 FMT_MOF.1(2) FMT_MOF.1(4) FMT_MOF.1(5) FMT_MOF.1(6) FMT_MOF.1(	FMT_MOF.1 restricts the ability to control the behavior of the audit and alarm mechanism to the administrators of the ASR. The Security Administrator is the only user that controls the behavior of the events that generate alarms and whether the alarm mechanism is enabled or disabled. FAU_SAR.2 restricts the ability to read the audit trail to the administrative users of the ASR, thus preventing the disclosure of the audit data to any other user. However, the TOE is not expected to prevent the disclosure of audit data if it has been archived or saved in another form (e.g., moved or copied to an ordinary file). The FAU_STG family dictates how the audit trail is protected. FAU_STG_(EXT).4 restricts the ability to delete audit records to the Audit Administrator; or if the option of overwriting old audit records is chosen by the Administrator in FAU_STG_(EXT).4, the audit data may be deleted/overwritten. Since the Audit Administrator is trusted to review the audit data, the threat being countered is that the administrator does something malicious and then attempts to conceal

	8)	it by configuring the audit log to overwrite old records. Presumably the administrator would then attempt to fill up the audit log in order to overwrite the thing they just did, and the fact that they reconfigured the audit log overwrite action. The Audit Administrator would hopefully notice this activity and detect the fact that the administrator was performing illicit activities. The fact that the administrator does not directly have the ability to delete the audit records helps ensure that audit records are kept until the Audit Administrator deems they are no longer necessary. FAU_STG_(EXT).4 also ensures that no one has the ability to modify audit records (e.g., edit any of the information contained in an audit record). This ensures the integrity of the audit trail is maintained. FAU_STG.3 requires that the administrators be alerted when the audit trail exceeds a capacity threshold established by the Security Administrator. In addition, an audit record is cut which will trigger the analysis performed in FAU_SAA, resulting in an FAU_ARP alarm being issued. This ensures that an administrator has the opportunity to manage the audit trail before it becomes full and the avoiding the possible loss of audit data. FAU_STG_(EXT).4 allows the Security Administrator to configure the TOE so that if the audit trail does become full, either the TOE will prevent any events from occurring (other than actions taken by the administrator) that would generate an audit record or the audit mechanism will overwrite the oldest audit records with new records. FMT_SMF.1 requires the TOE to provide an administrator with a facility to backup, recover and archive audit data ensuring the ability to recover corrupted audit records, and access to a complete history of audit information.
O.AUDIT_REVIEW	FAU_ARP.1 FAU_ARP_ACK_(EXT).1 FAU_SAA.1 FAU_SAR.1 FAU_SAR.3	FAU_SAA.1-NIAP-0407 defines the events (or rules) that indicate a potential security violation and will generate an alarm. The triggers for these events are largely configurable by the Security Administrator. Some rules are not configurable, or configurable by the cryptographic administrator. FAU_ARP.1 requires that the alarm be displayed at the local administrative console and at the remote administrative console(s) when auditor and security administrative session(s) exists. For alarms at remote consoles, the alarm is sent either during an established session or upon session establishment (as long as the alarm has not been acknowledged). This is required to increase the likelihood that the alarm will be received as soon as possible. This requirement also dictates the information that must be displayed with the alarm. The potential security violation is identified in the alarm, as are the contents of the audit records of the events that accumulated and triggered the alarm. The information in the audit records is necessary; it allows the administrators to react to the potential security violation without having to search through the audit trail looking for the related events. FAU_ARP_ACK_(EXT).1 requires that an alarm generated by the mechanism that implements the FAU_ARP requirement be maintained until an administrator acknowledges it. This ensures that the alarm message will not be obstructed and the administrators will be alerted of a potential security violation. Additionally, this requires that the acknowledgement be transmitted to users that received the alarm, thus ensuring that that set of administrators knows that the user specified in the acknowledgement message has addressed the alarm. FAU_SAR.1 (both iterations) is used to provide both the auditor and

		an external audit analysis function the capability to read the entire audit data contained in the audit trail. This requirement also mandates the audit information be presented in a manner that is suitable for the end user (auditor or external system) to interpret the audit trail. It is expected that the audit information be presented in such a way that the end user can examine an audit record and have the appropriate information (that required by FAU_GEN.2-NIAP-410) presented together to facilitate the analysis of the audit review. Ensuring the audit data are presented in an interpretable format will enhance the ability of the entity performing the analysis to identify potential security violations. 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. FAU_SAR.3 requires the administrators be able to establish the audit review criteria based on a userid and role so that the actions of a user can be readily identified and analyzed. Allowing the administrators to perform searches or sort the audit records based on dates and times provides the capability to facilitate the administrator's review of incidents that may have taken place at a certain time. It is important to note that the intent of sorting in this requirement is to allow the administrators the capability to organize or group the records associated with a given criteria.
O.CHANGE_MANAGEMENT	ALC_CMC.4 ALC_CMS.4 ALC_DVS.1 ALC_FLR.2 ALC_LCD.1	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_CSC.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 requires the developer describe the security measures they employ to ensure the integrity and confidentiality of the TOE is 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 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 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. ALC_CMC.4 and ALC_CMS.4 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.
O.CORRECT_ TSF_OPERATION	FPT_TST_(EXT).1, FPT_TST.1(1) FPT_TST.1(2)	O_CORRECT_TSF_OPERATION requires two security functional requirements in the FPT class, FPT_TST. These functional requirements provide the end user with the capability to ensure the TOE's security mechanisms continue to operate correctly in the field. FPT_TST_(EXT).1 has been created to ensure end user tests exist to demonstrate the correct operation of the security mechanisms required by the TOE that are provided by the hardware and that the TOE's software and TSF data has not been corrupted. Hardware failures could render a TOE's software ineffective in enforcing its security policies and this requirement provides the end user the ability to discover any failures in the hardware security mechanisms. FPT_TST.1(1) and FPT_TST.1(2) are necessary to ensure the correctness of the TSF software and TSF data. If TSF software is corrupted it is possible that the TSF would no longer be able to enforce the security policies. This also holds true for TSF data, if TSF data is corrupt the TOE may not correctly enforce its security policies.
O.CRYPTOGRAPHIC _FUNCTIONS	FCS_BCM_(EXT).1 FCS_CKM.1(1) FCS_CKM.1(2) FCS_CKM.2 FCS_CKM.4 FCS_CKM_(EXT).2 FCS_COP.1(1) FCS_COP.1(2) FCS_COP.1(3) FCS_COP.1(4) FCS_COP.1(5)	The FCS requirements used in this PP satisfy this objective by levying requirements that ensure the cryptographic standards include the NIST FIPS publications (where possible) and NIST approved ANSI standards. The intent is to have the satisfaction of the cryptographic standards be validated through a NIST FIPS 140 validation. The core functionality to be supported is encryption/decryption using a symmetric algorithm, and digital signature generation and verification using asymmetric algorithms. Since these operations involve cryptographic keys, how the keys are generated and/or otherwise obtained have to also be specified. FCS_BCM_(EXT).1 is an extended requirement that specifies not only that cryptographic functions that are FIPS-approved and must be validated by FIPS, but also what NIST FIPS rating level the cryptographic module must satisfy. The level specifies the degree of testing of the module. The higher the level, the more extensive the module is tested. FCS_CKM.1(1) is a requirement that a cryptomodule generate symmetric keys. Such keys are used by the TDEA or AES encryption/decryption functionality specified in FCS_COP.1(1). FCS_CKM.1(2) is a requirement that a cryptomodule generate asymmetric keys. Such keys are used for cryptographic signatures as specified in FCS_COP.1(2). FCS_CKM.1 requires that the TSF validate all keys generated to assure that it meet relevant standards. FCS_CKM_(EXT).2 requires that keys are handled appropriately and associated with the correct entities, and that transfer of keys is done with error detection. Storage of persistent secret and private keys must be done in a secure fashion. FCS_COP.1(3) requires that the TSF provide hashing services using a NIST-approved implementation of the Secure Hash Algorithm and FCS_COP.1(4) requires the TSF's message authentication services be

		compliant with either of the NIST-approved approaches, HMAC or CCM. Another way of obtaining key material for symmetric algorithms is through cryptographic key establishment, as specified in FCS_COP.1(5). Key establishment has two aspects: key agreement and key distribution. Key agreement occurs when two entities exchange public data yet arrive at a mutually shared key without ever passing that key between the two entities (for example, the Diffie-Hellman algorithm). Key distribution (FCS_CKM.2) occurs when the key is transmitted from one entity to the TOE. If the entity is electronic and a protocol is used to distribute the key, it is referred to in this PP as "Key Transport". If the key is loaded into the TOE it can be loaded electronically (e.g., from a floppy drive, smart card, or electronic keyfill device) or manually (e.g., typed in). One or more of these methods must be selected. FCS_CKM.4 provides the functionality for ensuring key and key material is zeroized. This applies not only to key that resides in the TOE, but also to intermediate areas (physical memory, page files, memory dumps, etc.) where key may appear. FCS_COP.1(1) specifies that TDEA or AES be used to perform encryption and decryption operations. FCS_COP.1(2) gives three options for providing the digital signature capability; these requirements reference the appropriate standards for each digital signature option.
	FCS_COP_(EXT).1	FCS_COP_(EXT).1 requires that any random number generation and hashing functions, respectively, are part of a FIPS-validated cryptographic module. This requirement does not mandate that the functionality is generally available, but only that it be implemented in a FIPS-validated module should other cryptographic functions need these services.
O.CRYPTOGRAPHY_VALIDATED	FCS_BCM_(EXT).1 FCS_CKM.1(1) FCS_CKM.1(2)	This objective deals with the issue of using FIPS 140-2-approved cryptomodules in the TOE. A cryptomodule, as used in the components, is a module that is FIPS 140-2 validated (in accordance with FCS_BCM_(EXT).1); the cryptographic functionality implemented in that module are FIPS-approved security functions that have been validated; and the cryptographic functionality is available in a FIPS-approved mode of the cryptomodule. This objective is distinguished from O.CRYPTOGRAPHIC_FUNCTIONS in that this deals only with a requirement to use FIPS 140-2-validated cryptomodules where the TOE requires such functionality; it does not dictate the specific functionality that is to be used. FCS_BCM_(EXT).1 is an extended requirement that specifies not only that cryptographic functions that are FIPS-approved must be validated by FIPS, but also what NIST FIPS rating level the cryptographic module must satisfy. The level specifies the degree of testing of the module. The higher the level, the more extensive the module is tested. FCS_CKM.1(1) and FCS_CKM.1(2) mandates that the cryptomodule must generate key, and that this key generation must be part of the FIPS-validated cryptomodule.
O.DISPLAY_BANNER	FTA_TAB.1	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.HA	FPT_HA_(EXT).1	FPT_HA_(EXT).1 requires that the TOE provide failover capability for configurations that include dual ESPs or RPs.
O.INTEGRITY	FDP_IFC.1(1) FDP_IFF.1(1)	O.INTEGRITY (FDP_IFC.1(1), FDP_IFF.1(1)) satisfies this policy by ensuring that all IPSec encrypted data received from a peer TOE is properly decrypted and authentication verified.
O.MAINT_MODE	FPT_RCV.1 FPT_RCV.2	This objective is met by using the FPT_RCV.2 and FPT_RCV.1 requirements, which ensures that the TOE does not continue to operate in an insecure state when a hardware or software failure occurs. Upon the failure of the TSF self-tests the TOE will no longer be assured of enforcing its security policies. Therefore, the TOE enters a state that operations cease and requires an administrator to follow documented procedures that instruct them on to return the TOE to a secure state. These procedures may include running diagnostics of the hardware, or utilities that may correct any integrity problems found with the TSF data or code. Solely specifying that the administrator reload and install the TOE software from scratch, while may be required in some cases, does not meet the intent of this requirement.
O.MANAGE_INFORMATION_FLOW (AKA O.MANAGE)	FMT_MSA.1(1) FMT_MSA.1(2) FMT_MSA.3(1) FMT_MSA.3(2) FMT_MSA.3(3) FMT_MOF.1(1) FMT_MOF.1(2) FMT_MOF.1(3) FMT_MOF.1(4) FMT_MOF.1(5) FMT_MOF.1(6) FMT_MOF.1(7) FMT_MOF.1(8) FMT_MTD.1(1) FMT_MTD.1(2) FMT_MTD.1(3) FMT_MTD.1(4) FMT_SMF.1	The FMT requirements are used to satisfy this management objective, and 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.1 all provide the Security Administrator the capability to manipulate the security attributes of the objects in their scope of control that determine the access policy. FMT_MSA.3(1) requires that by default, the TOE does not allow an information flow, rather than allowing information flows until a rule in the ruleset disallows it. FMT_MOF.1(2) and FMT_MSA.3(2) are related to the services provided by FAU_UAU.1 and provide the Security Administrator control as to the availability of these services. FMT_MOF.1(2) provides the ability to enable or disable the TOE services to the Security Administrator. FMT_MSA.3(2) requires that these services by default are disabled. Since the Security Administrator must explicitly enable these services it ensures the Security Administrator is aware that they are running. This requirement does afford the Security Administrator the capability to override this restrictive default and allow the services to be started whenever the TOE reboots or is restarted. FMT_MSA.3(3) requires that these services by default are disabled. Since the Security Administrator must explicitly enable these services it ensures the Security Administrator is aware that they are running. This requirement does afford the Security Administrator the capability to override this restrictive default and allow the services to be started whenever the TOE reboots or is restarted. FMT_MOF.1(1) is used to ensure the administrators have the ability to invoke the TOE self-tests at any time. The ability to invoke the self-tests is provided to all administrators. The Security Administrator is able to modify the behavior of the tests (e.g., select when they run, select a subset of the tests). FMT_MOF.1(3) specifies the ability of the administrators to control the security functions associated with audit and alarm generation. The ability to control these functions has been assigned to the appropriate

		administrative roles. FMT_MOF.1(6) This requirement limits the ability to manipulate the values that are used in the FRU_RSA.1(2) requirements to the Security Administrator. The Security Administrator is provided the capability to assign the network identifier(s) they wish to place resource restrictions on and allows them to also specify over what period of time those quota limitations are in place. FMT_MOF.1(4) provides the administrators "read only" access to the audit records and prohibits access to all other users. Additionally, the administrators are provided the capability to "search and sort" audit on defined criteria. This capability expedites problem resolution analysis. FMT_MOF.1(5) ensures that only an administrators can "enable or disable" the security alarms. This requirement works with FMT_MOF.1(5) to provide detailed granularity to the administrator when determining which actions constitute a security violation. FMT_MOF.1(6) provides the Security Administration configuration control of the allocation of connection-oriented TOE resources. This requirement provides the Security Administrator with a capability to thwart possible external "resource allocation" attacks on the TOE. FMT_MOF.1(7) provides the Security Administration configuration control of unsuccessful authentication attempts The requirement FMT_MTD.1(1) is intended to be used by the ST author, with possible iterations, to address TSF data that has not already been specified by other FMT requirements. This is necessary because the ST author may add TSF data in assignments that cannot be addressed ahead of time by the PP authors. This requirement specifies that the manipulation of these data be restricted to the security administrator. FMT_MOF.1(8) ensures that only an administrators can "enable, disable, determine and modify the behavior" the failure handling. FMT_MTD.1(2) provides the Cryptographic Administrator, and only the Cryptographic Administrator, the ability to modify the cryptographic security data. This allows the Cryptographic Administrator to change the critical data that affects the TOE's ability to perform its cryptographic functions properly. FMT_MTD.1(3) provides the capability of setting the date and time that is used to generate time stamps to the Security Administrator. It is important to allow this functionality, due to clock drift and other circumstances, but the capability must be restricted. FMT_MTD.1(4) addresses the capabilities of data managers, who have responsibilities for security data management for sub-portions of the set of TSF data (for example, the platform clock time, sub-hierarchies of the directory). The scope of a data manager's responsibility is set by a security administrator, but they are expected to manage the entities in their scope of control without reliance on the security administrator. FMT_MTD.2(1), FMT_MTD.2(2), FMT_MTD.2(3) restrict the setting of limits on the processor time, network connection resources and audit storage limits respectively, to an administrator. This capability allows an administrator to control the resources consumed by, to provide a flexible policy with respect to denial of service attacks. The requirement FMT_SMF.1 was introduced as an international interpretation. This requirement specifies functionality that must be provided to administrators of the TOE. If the PP author includes this

		requirement, care must be taken to use the other FMT requirements to specify how the functionality is restricted and to which role the functionality is provided.
O.MEDIATE_INFORMATION_FLOW (AKA O.MEDIATE)	FDP_IFC.1(1) FDP_IFC.1(2) FDP_IFC.1(4) FDP_IFF.1(1) FDP_IFF.1(2) FDP_IFF.1(4 FMT_REV.1 ADV_ARC.1	The FDP_IFF and FDP_IFC requirements were chosen to define the policies, the subjects, objects, and operations for how and when mediation takes place. FDP_IFC.1(1) - (4) define the subjects, information (e.g., objects) and the operations that are performed with respect to the two information flow policies. FDP_IFC.1(1), the subjects are the TOE's network interfaces. The objects are defined as the network IP packets on which the TOE performs VPN operations. As packets enter the TOE, the network interface where they are received is the source subject. As packets are sent out of the TOE the network interface that they are sent out of is the destination subject. Subjects must be defined as entities that the TOE has control over. The TOE has control over its own network interfaces such that it can make information flow decisions to allow/disallow network packets to flow from in incoming interface to an outgoing interface, and can apply VPN operations to packets that are allowed to flow. To define subjects as the senders and receivers of network packets would not allow specification of an information flow policy that the TOE could enforce, since the sender and receiver of network packets are not under the control of the TOE. The operations defined are those of the VPN policy. The VPN policy either passes information unmodified, sends encrypted and authenticated packets to a peer TOE, or decrypts and verifies authentication of packets received from a peer TOE. FDP_IFF.1(1) specifies the attributes on which VPN information flow decisions are made. Each TOE interface has a set of source subject identifiers that is the list of senders of information packets that are allowed to send packets to this TOE interface. Each TOE interface also has a list of destination subject identifiers that specifies the receivers that network packets can be sent to on that TOE interface. As packets are received on a particular network interface, the TOE determines if they are allowed to enter on that interface. Then based on rules defined by the Security Administrator, the TOE applies VPN operations to the packet. Before the packet is sent out of a particular network interface, the TOE determines if the destination (i.e., receiver) of the packet is in the list of destinations that may be reached over that interface. FDP_IFC.1(2) defines subjects for the unauthenticated access to any services the TOE provides. This is different from the other policies in that the TOE mediates access to itself, rather than determining if information should be allowed to flow through the TOE. The destination subject is defined to be the TOE, and the source subject is the TOE interface on which a network packet is received. The information remains the same, a network packet, and the operations are limited to accept or reject the packet. FDP_IFF.1(2) provides the rules that apply to the unauthenticated use of any services provided by the TOE. ICMP is the only service that is required to be provided by the TOE, and the security attributes associated with this protocol allow the Security Administrator to specify what degree the ICMP traffic is mediated (i.e., the ICMP message type and code). The ST author could specify other services they wish their TOE implementation to provide, and if they do so, they should also specify the security attributes associated with the

		additional services. FMT_REV.1 is a management requirement that affords the Security Administrator the ability to immediately revoke user's ability to send network traffic to or through the TOE. FDP_IFC.1(4), the subjects are defined to be a source subject, which is the TOE's network interface on which a packet is received, and a destination subject, which is the TOE's network interface on which the packet is destined. The information flow control requirements are not well suited for a firewall. This subject determination was made since the TOE network interfaces are something the TOE has control over (e.g., the administrator has the ability to assign network identifiers to these interfaces, which is a critical component in the mediation decision) and rules could be identified in FDP_IFF.1(4) that make sense with respect to mediation of information. The alternative was to classify the sender and receiver of the data packets as subjects, but the sender and receiver are not under the control of the TOE and would not make sense to perform mediation under those circumstances. The objects in this policy are defined to be the network packets, since that is the entity that the operations are performed on. Those operations are to pass the information if the mediation allows the flow, otherwise the packet is dropped. FDP_IFF.1(4) is used to specify the policy of unauthenticated traffic flowing through the TOE. This requirement ensures that the network traffic is mediated (i.e., the ruleset is used) even though the subjects have not been authenticated. This requirement also mandates the TOE perform stateful inspection of the packets to determine if they should be allowed to flow through the TOE. The stateful inspection attributes are not intended to specifiable by the Security Administrator, rather these attributes are to be "managed" and mediated internally by the TOE. FMT_REV.1 is a management requirement that affords the Security Administrator the ability to immediately revoke user's ability to send network traffic to or 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 describes an architecture that ensures packets that flow through the TOE, or those that are destined for the TOE are mediated with respect to the identified policies. Each TSF interface that operates on subjects or objects that are identified in the explicit policies, or operates on TSF data or security attributes, must ensure that the operation is checked against the explicit and implicit security policies defined in this PP. If any TSF interface allows unchecked access to any of these resources, then the TOE cannot be relied upon to enforce the security policies. 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. This will ensure that packets that flow through the TOE, or those that are destined for the TOE are mediated with respect to the identified policies. Each TSF interface that operates on subjects or objects that are identified in the explicit policies, or operates on TSF data or security attributes, must ensure that the operation is checked against the explicit and implicit security policies defined in this PP. If any TSF interface allows unchecked

		access to any of these resources, then the TOE cannot be relied upon to enforce the security policies.
O.PEER_AUTHENTICATION	FCS_IKE_(EXT).1	The O.PEER_AUTHENTICATION objective is satisfied by the requirement FCS_IKE_(EXT).1, which specifies that the TOE must implement the Internet Key Exchange protocol defined in RFC 2409. By implementing this protocol, the TOE will establish a secure, authenticated channel with each peer TOE for purposes of establishing a security association, which includes the establishment of a cryptographic key, algorithm and mode to be used for all communication. It is possible to establish multiple security associations between two peer TOEs, each with its own cryptographic key. Authentication may be via a digital signature or pre-shared key.
	FCS_GDOI_(EXT).1	The O.PEER_AUTHENTICATION objective is satisfied by the requirement FCS_GDOI_(EXT).1, which specifies that the TOE must implement the Group Domain of Interpretation protocol defined in RFC 3547. By implementing this protocol, the TOE will establish a secure, authenticated channel with groups of peer TOEs for purposes of establishing a security association, which includes the establishment of a cryptographic key, algorithm and mode to be used for all communication.
O.PROTECT_IN_TRANSIT	FPT_ITA.1 FPT_ITC.1 FPT_ITI.1 FTP_ITC.1(1) FTP_ITC.1(2) FTP_TRP.1(1) FTP_TRP.1(2)	FPT_ITA.1, FPT_ITC.1 and FPT_ITI.1 are concerned with the availability, confidentiality and integrity of the TSF data while being transmitted. FTP_ITC.1(1) and FTP_ITC.1(2) ensures that all TSF data will be protected from disclosure while in transit from the TOE to another trusted IT entity. FTP_TRP.1(1) and FTP_TRP.1(2) will use cryptographic means to provide prevention of disclosure and detection of modification of TSF data.
O.PROTOCOLS	FPT_FLS.1 FPT_PRO_(EXT).1	The O.PROTOCOLS objective is satisfied by FPT_PRO_(EXT).1, which requires that the TOE be implemented with standardized protocols to ensure interoperability among peer TOEs. Implementing the standardized protocols will ensure that a secure state (FPT_FLS.1) of the TOE is maintained.
O.REPLAY_DETECTION	FPT_RPL.1	The O.REPLAY_DETECTION objective is satisfied by FPT_RPL.1, which requires the TOE to detect and reject the attempted replay of authentication data from a remote user. This is sufficient to meet the objective because no untrusted users have local access to the TOE, thus there is no way to capture neither replay authentication data for a local session.
O.RESIDUAL_INFORMATION	FDP_RIP.2 FCS_CKM.4 FCS_CKM_(EXT).2	FDP_RIP.2 is used to ensure the contents of resources are not available to subjects other than those explicitly granted access to the data. For this TOE it is critical that the memory used to build network packets is either cleared or that some buffer management scheme be employed to prevent the contents of a packet being disclosed in a subsequent packet (e.g., if padding is used in the construction of a packet, it must not contain another user's data or TSF data). 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.
O.RESOURCE_SHARING	FRU_RSA.1(1) FRU_RSA.1(2	While an availability security policy does not explicitly exist, FRU_RSA.1 was used to mitigate potential resource exhaustion attempts. FRU_RSA.1(1) was used to reduce the impact of an attempt

	) FMT_MTD.2(1) FMT_MTD.2(2) FMT_MOF.1(6) FMT_MOF.1(7)	being made to exhaust the transport-layer representation (e.g., attempt to make the TSF unable to respond to connection-oriented requests, such as SYN attacks). This requirement allows the administrator to specify the time period in which when maximum quota (which is defined by the ST) is met or surpassed, an ST defined action is to take place, which is specified in FMT_MTD.2(1). These two requirements together help limit the resources that can be utilized by the general population of users as a whole. An issue with treating all the users the same is that legitimate users may not be able to establish connections due to the connection table entries being exhausted. Therefore FRU_RSA.1(2) is also included. FRU_RSA.1(2) is more specific in that attempts to exhaust the connection-oriented resources by a single network address, or a set of network addresses can be controlled. This affords the administrator a finer granularity of control than FRU_RSA.1(1). FRU_RSA.1(2) has the advantage of providing the Security Administrator with the ability to define the maximum number of resources a particular address or set of addresses can use over a specified time period. This requirement works in conjunction with FMT_MTD.2(2) which restricts the ability to set the quotas to the security administrator and allows for the ST author to assign what actions will take place once the quotas are met or surpassed. This iteration of FPT_RSA.1 makes it less likely that a legitimate user of the TOE will be denied access due to resource exhaustion attempts. FMT_MOF.1(6) and (7) restricts the ability to assign the single network address or set of network addresses used in FRU_RSA.1(2) to the Security Administrator. This is in keeping with the TOE's notion of the Security Administrator is responsible for configuring the TOE's policy enforcement mechanisms.
O.ROBUST_ADMIN_GUIDANCE	ALC_DEL.1 AGD_PRE.1 AGD_OPE.1	ALC_DEL.1 ensures that the administrator is provided documentation that instructs them how to ensure the delivery of the TOE, in whole or in parts, has not been tampered with or corrupted during delivery. This requirement ensures the administrator has the ability to begin their TOE installation with a clean (e.g., malicious code has not been inserted once it has left the developer's control) version of the TOE, which is necessary for secure management of the TOE. The AGD_PRE.1 requirement ensures the administrator has the information necessary to install the TOE in the evaluated configuration. Often times a vendor's product contains software that is not part of the TOE and has not been evaluated. The Operational Users Guidance and the Preparative procedures ensure that once the administrator has followed the installation and configuration guidance the result is a TOE in a secure configuration. The AGD_OPE.1 requirement mandates the developer provide the administrator with guidance on how to operate the TOE in a secure manner. This includes describing the interfaces the administrator uses in managing the TOE, security parameters that are configurable by the administrator, how to configure the TOE's ruleset and the implications of any dependencies of individual rules. The documentation also provides a description of how to setup and review the auditing features of the TOE. The AGD_OPE.1 is also intended for non-administrative users, but could be used to provide guidance on security that is common to both administrators and non-administrators (e.g., password management guidelines).

		FIA_UAU.2 requires that administrators and authorized IT entities authenticate themselves to the TOE before performing any TSF-mediated actions. In order to control logical access to the TOE an authentication mechanism is required. The extended requirement FIA_UAU_(EXT).5 mandates that the TOE provide a local authentication mechanism. This requirement also affords the ST author the opportunity to add additional authentication mechanisms (e.g., single-use, certificates) if they desire. Local authentication is required to ensure someone that has physical access to the TOE and has not been granted logical access (e.g., a janitor) cannot gain unauthorized logical access to the TOE. 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 and authorized IT entities can access the TOE. 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. The FTA_SSL family partially satisfies the O.ROBUST_TOE_ACCESS objective by ensuring that user's sessions are afforded some level of protection. FTA_SSL.3 takes into account remote sessions. After a Security Administrator defined time interval of inactivity remote sessions will be terminated. This includes user remote administrative sessions. This component is especially necessary; since remote sessions are not typically afforded the same physical protections those local sessions are provided.
O.ROBUST_TOE_ACCESS	FIA_AFL.1 FIA_ATD.1(1) FIA_ATD.1(2) FIA_UAU.1 FIA_UAU.2 FIA_UAU_(EXT).5 FIA_UID.2 FTA_SSL.1 FTA_SSL.2 FTA_SSL.3 FTA_TSE.1 AVA_VAN.3	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., proxy users, 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_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 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.2 was refined since only the VPN only requires that administrators, authorized IT entities and proxy users authenticate themselves to the TOE before performing administrative duties or using the proxy services identified in this requirement. Unlike the unauthenticated proxies, these proxies require authentication, which provides a level of control on who can access the proxies and reduces the potential risk to the TOE. In order to control logical access to the TOE an authentication mechanism is required. The extended requirement FIA_UAU_(EXT).5 mandates that the TOE provide a local authentication mechanism. This requirement also affords the ST author the opportunity to add additional authentication mechanisms (e.g., single-use, certificates) if they desire. Local authentication is required to ensure someone that has physical access to the TOE and has not been granted logical access (e.g., a janitor) cannot gain unauthorized logical access to the TOE. The AVA_VAN.3 requirement as applied to the local authentication mechanism. 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, as defined in Annex B of the CEM. 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 proxy users and authorized IT entities can access the TOE. FIA_AFL.1 provides a detection mechanism for unsuccessful authentication attempts by remote administrators, authenticated proxy users 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. The FTA_SSL family partially satisfies the O.ROBUST_TOE_ACCESS objective by ensuring that user's sessions are afforded some level of protection. FTA_SSL.1 provides the Security Administrator the capability to specify a time interval of inactivity in which an unattended local administrative session would be locked and will require the administrator responsible for that session to re-authenticate before the session can be used to access TOE resources. FTA_SSL.2 provides administrators the ability to lock their local administrative session. This component allows administrators to protect their session immediately, rather than waiting for the time-out period and minimizes their session's risk of

		exposure. FTA_SSL.3 takes into account remote sessions. After a Security Administrator defined time interval of inactivity remote sessions will be terminated, this includes user proxy sessions and remote administrative sessions. This component is especially necessary, since remote sessions are not typically afforded the same physical protections that local sessions are provided.
O.SELF_PROTECTION	ADV_ARC.1	ADV_ARC.1 will require an architecture that ensure the TSF provides a domain that protects itself from untrusted users. If the TSF cannot protect itself it cannot be relied upon to enforce its security policies. ADV_ARC.1 also provides an architecture to ensure that the cryptographic module be provided its own address space. This is necessary to reduce the impact of programming errors in the remaining portions of the TSF on the cryptographic module. The inclusion of ADV_ARC.1 ensures that the TSF makes policy decisions on all interfaces that perform operations on subjects and objects that are scoped by the policies. Without this non-bypassability requirement, the TSF could not be relied upon to completely enforce the security policies, since an interface(s) may otherwise exist that would provide a user with access to TOE resources (including TSF data and executable code) regardless of the defined policies. This includes controlling the accessibility to interfaces, and what access control is provided within the interfaces.
	FTP_ITC.1(1) FTP_ITC.1(2) FTP_TRP.1(1) FTP_TRP.1(2)	FTP_ITC.1(1), FTP_ITC.1(2) and FTP_TRP.1(1), FTP_TRP.1(2) are necessary for communication between the TOE and other trusted IT entities (e.g., authentication server, authorized IT entities) and the TOE and remote administrators. In order to protect TSF data and security attributes there is need for a trusted channel/trusted path. The trusted channel ensures that the authentication data that is supplied to the TOE is not compromised. It may be the case that the TOE relies upon an authorized IT entity to supply/manage TSF data (e.g., time stamp). If this is the case, the trusted channel ensures the TSF data is not compromised. The aspect of the trusted path that applies to this objective is FTP_TRP.1.3, which requires that the entire remote administrative session be protected. The protection of the communication path when TSF data is being transmitted is critical to the TSF maintaining a domain of execution that cannot be tampered or interfered with, thus resulting is a possible unauthorized disclosure or security policy failure.
O.SOUND_DESIGN	ADV_ARC.1 ADV_FSP.4 ADV_TDS.3	There are two different perspectives for this objective. One is from the developer's point of view and the other is from the evaluator's. The ADV class of requirements is levied to aide in the understanding of the design for both parties, which ultimately helps to ensure the design is sound. ADV_ARC.1 addresses the non-bypassability and domain separation aspects of the TSF, since this needs to be analyzed differently from other requirements. The low-level design, as required by ADV_TDS.3, provides the reader with the details of the TOE's

		design and describes at a module level how the design of the TOE addresses the SFRs. This level of description provides the detail of how modules interact within the TOE and if a flaw exists in the TOE's design; it is more likely to be found here rather than the high-level design. This requirement also mandates that the interfaces presented by modules be specified. Having knowledge of the parameters a module accepts, the errors that can be returned and a description of how the module works to support the security policies allows the design to be understood at its lowest level. ATE_DPT.1AVA_VAN.3ADV_FSP.4 requires that the interfaces to the TSF be completely specified. In this TOE, a complete specification of the network interface (including the network interface card) is critical in understanding what functionality is presented to untrusted users and how that functionality fits into the enforcement of security policies. Some network protocols have inherent flaws and users have the ability to provide the TOE with network packets crafted to take advantage of these flaws. The routines/functions that process the fields in the network protocols allowed (e.g., TCP, UPD, ICMP, any application level) must fully specified: the acceptable parameters, the errors that can be generated, and what, if any, exceptions exist in the processing. The functional specification of the hardware interface (e.g., network interface card) is also extremely critical. 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 requires that design of the TOE be provided. This level of design describes the architecture of the TOE in terms of subsystems. It identifies which subsystems are responsible for making and enforcing security relevant (e.g., anything relating to an SFR) decisions and provides a description, at a high level, of how those decisions are made and enforced. Having this level of description helps provide a general understanding of how the TOE works, without getting buried in details, and may allow the reader to discover flaws in the design. ADV_TDS.3 also provides the reader with the details of the TOE's design and describes at a module level how the design of the TOE addresses the SFRs. This level of description provides the detail of how modules interact within the TOE and if a flaw exists in the TOE's design, it is more likely to be found here rather than the high-level design. This requirement also mandates that the interfaces presented by modules be specified. Having knowledge of the parameters a module accepts, the errors that can be returned and a description of how the module works to support the security policies allows the design to be understood at its lowest level.
O.SOUND_IMPLEMENTATION	ADV_IMP.1 ADV_ARC.1 ADV_TDS.3 ALC_TAT.1	While ADV_TDS.3 is used to aide in ensuring that the TOE's design is sound, it also contributes to ensuring the implementation is correctly realized from the design. It is expected that evaluators will use the low-level design as an aide in understanding the implementation representation. The low-level design requirements ensure the evaluators have enough information to intelligently analyze (e.g., the documented interface descriptions of the modules match the entry points in the module, error codes returned by the functions in the module are consistent with those identified in the documentation) the implementation and ensure it is consistent with the design.

		ADV_IMP.2 was chosen to ensure evaluators have full access to the source code. If the evaluators are limited in their ability to analyze source code they may not be able to determine the accuracy of the implementation or the adequacy of the documentation. Often times it is difficult for an evaluator to identify the complete sample of code they wish to analyze. Often times looking at code in one subsystem may lead the evaluator to discover code they should look at in another subsystem. Rather than require the evaluator to "re-negotiate" another sample of code, the complete implementation representation is required. ALC_TAT.1 provides evaluators with information necessary to understand the implementation representation and what the resulting implementation will consist of. Critical areas (e.g., the use of libraries, what definitions are used, compiler options) are documented so the evaluator can determine how the implementation representation is to be analyzed.
O.THOROUGH_ FUNCTIONAL_TESTING	ATE_COV.3 ATE_FUN.1 ATE_DPT.1 ATE_IND.2	In order to satisfy O.FUNCTIONAL_TESTING, the ATE class of requirements is necessary. The component 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. In addition, the developer must provide the test suite executables and source code, which are used for independently verifying the test suite results and in support of the test coverage analysis activities. ATE_COV.3 requires the developer to provide a test coverage analysis that demonstrates the TSFI are completely addressed by the developer's test suite. While exhaustive testing of the TSFI is not required, this component ensures that the security functionality of each TSFI is addressed. This component also requires an independent confirmation of the completeness of the test suite, which aids in ensuring that correct security relevant functionality of a TSFI is demonstrated through the testing effort. ATE_DPT.1 requires the developer to provide a test coverage analysis that demonstrates depth of coverage of the test suite. This component complements ATE_COV.3 by ensuring that the developer takes into account the high-level and low-level design when developing their test suite. Since exhaustive testing of the TSFI is not required, ATE_DPT.1 ensures that subtleties in TSF behavior that are not readily apparent in the functional specification are addressed in the test suite. 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. This component also requires an independent party to attempt to craft functional tests that address functional behavior that is not demonstrated in the developer's test suite. Upon successful adherence to these requirements, the TOE's conformance to the specified security functional requirements will have been demonstrated.
O.TIME_STAMPS	FPT_STM.1 FMT_MTD.1(3)	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(3) satisfies the rest of this objective by providing the capability to set the time used for generating time stamps to either the Security Administrator, authorized IT entity, or both, depending on the selection made by the ST author.
O.TRUSTED_PATH	FTP_ITC.1(1),	FTP_TRP.1.1 requires the TOE to provide a mechanism that creates a

	FTP_ITC.1(2) FTP_TRP.1(1) , FTP_TRP.1(2)	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. FTP_ITC.1(1) and FTP_ITC.1(2) are similar to FTP_TRP.1(1) and FTP_TRP.1(2), in that they require a mechanism that creates a distinct communication path with the same characteristics, however FTP_ITC.1(1) and FTP_ITC.1(2) 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.
O.USER_GUIDANCE	AGD_PRE.1	O.USER_GUIDANCE (AGD_PRE.1) mitigates this threat by providing the user the information necessary to use the security mechanisms that control access to user data in a secure manner. For instance, the method by which the discretionary access control mechanism (FDP_ACC.1, FDP_ACF.1) is configured, and how to apply it to the data the user owns, is described in the user guidance. If this information were not available to the user, the information may be left unprotected, or the user may mis-configure the controls and unintentionally allow unauthorized access to their data.
O.VULNERABILITY_ANALYSIS_TEST	AVA_VAN.3	To maintain consistency with the overall assurance goals of this TOE, O.VULNERABILITY_ANALYSIS_TEST requires the AVA_VAN.3 component to provide 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 developer to perform a systematic search for potential vulnerabilities in all the TOE deliverables. For those vulnerabilities that are not eliminated, a rationale must be provided that describes why these vulnerabilities cannot be exploited by a threat agent with a moderate attack potential, which is in keeping with the desired assurance level of this TOE. As with the functional testing, a key element in this component is that an independent assessment of the completeness of the developer's analysis is made, and more importantly, an independent vulnerability analysis coupled with testing of the TOE is performed. This component provides the confidence that security flaws do not exist in the TOE that could be exploited by a threat agent of moderate (or lower) attack potential to violate the TOE's security policies.

7.4 RATIONALE FOR THE SECURITY OBJECTIVES AND SECURITY FUNCTIONAL REQUIREMENTS FOR THE ENVIRONMENT

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

All but one of the security objectives for the environment, OE.CRYPTANALYTIC, are restatements of an assumption found in Section 3. Therefore, those security objectives for the non-IT environment trace to the assumptions trivially and are suitable for covering the assumptions.

The IT environment security objective OE.CRYPTANALYTIC is necessary to play a role in countering the threat T.CRYPTO_COMPROMISE. This IT environment security objective ensures that the cryptographic methods used in the IT environment are interoperable with the mechanisms provided by the TOE. The IT environment's cryptographic methods should be independently validated to be FIPS 140-2 compliant. OE.CRYPTANALYTIC maps to the IT environmental iterated requirements FPT_ITC.1 (ensuring that encryption is used on the communication channel between authorized IT entities and the TOE), and FPT_TRP (ensuring that an administrator and authenticated proxy users can be assured that they are communicating with the TOE).

ANNEX A: REFERENCES

7.5 References

The following documentation was used to prepare this ST:

Table 31: References

[CC_PART1]	Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and general model, dated July 2009, version 3.1, Revision 3, CCMB-2009-07-001
[CC_PART2]	Common Criteria for Information Technology Security Evaluation – Part 2: Security functional components, dated July 2009, version 3.1, Revision 3, CCMB--2009-07-002
[CC_PART3]	Common Criteria for Information Technology Security Evaluation – Part 3: Security assurance components, dated July 2009, version 3.1, Revision 3, CCMB-2009-07-003
[CEM]	Common Methodology for Information Technology Security Evaluation – Evaluation Methodology, dated July 2009, version 3.1, Revision 3, CCMB-2009-07-004