X-ES Xpedite5205 Embedded Services Router

Security Target

Version 1.0

October 13, 2014

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Table of Contents

List of Tables

List of Figures

List of Acronyms

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

Table 1 Acronyms

Acronyms / Abbreviations	Definition
AAA	Administration, Authorization, and Accounting
ACL	Access Control Lists
AES	Advanced Encryption Standard
BRI	Basic Rate Interface
CC	Common Criteria for Information Technology Security Evaluation
CEM	Common Evaluation Methodology for Information Technology Security
CM	Configuration Management
CSU	Channel Service Unit
DHCP	Dynamic Host Configuration Protocol
DSU	Data Service Unit
EAL	Evaluation Assurance Level
EHWIC	Ethernet High-Speed WIC
ESR	Embedded Services Router
GE	Gigabit Ethernet port
HTTP	Hyper-Text Transport Protocol
HTTPS	Hyper-Text Transport Protocol Secure
ICMP	Internet Control Message Protocol
ISDN	Integrated Services Digital Network
IT	Information Technology
NDPP	Network Device Protection Profile
OS	Operating System
PoE	Power over Ethernet
POP3	Post Office Protocol
PP	Protection Profile
SFP	Small–form-factor pluggable port
SHS	Secure Hash Standard
SIP	Session Initiation Protocol
SSHv2	Secure Shell (version 2)
ST	Security Target
TCP	Transport Control Protocol
TOE	Target of Evaluation
TSC	TSF Scope of Control
TSF	TOE Security Function
TSP	TOE Security Policy
UDP	User datagram protocol
VTY	Virtual terminal
WAN	Wide Area Network
WIC	WAN Interface Card

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 X-ES Xpedite5205 Embedded Services Router (ESR). 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. Administrators of the TOE will be referred to as administrators, Authorized Administrators, TOE administrators, semi-privileged, privileged administrators, and security administrators in this document. The Common Criteria Functional Specification is met through the description of interfaces in this Security Target and the parameters described within the Common Criteria Guidance Documentation as well as the Cisco documentation for TOE.

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 2.

1.1 ST and TOE Reference

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

Table 2 ST and TOE Identification

Name	Description
ST Title	X-ES Xpedite5205 Embedded Services Router
ST Version	1.0
Publication Date	October 13, 2014
Vendor and ST Author	Cisco Systems, Inc.
TOE Reference	X-ES Xpedite5205 Embedded Services Router
TOE Hardware Models	X-ES XPedite5205
TOE Software Version	IOS 15.2(4)GC
ST Evaluation Status	In Evaluation
Keywords	Router, Data Protection, Authentication, Firewall

1.2 TOE Overview

The Cisco X-ES Xpedite5205 Embedded Services Router (ESR) running IOS 15.2(4)GC (herein after referred to as the XPedite5205, the router, or the TOE). The TOE is a high-performance, ruggedized router designed for use in harsh environments-offering reliable operation in extreme temperatures and under shock and vibration conditions typical for mobile applications in rugged terrain.

1.2.1 TOE Product Type

The Cisco X-ES Xpedite5205 Embedded Services Router is a router platform used to construct IP networks by interconnecting multiple smaller networks or network segments. The TOE provides connectivity and security services onto a single, secure device. The flexible, compact form factor of these routers, complemented by Cisco IOS® Software, provides highly secure data, voice, and video communications to stationary and mobile network nodes across wired links.

In support of the routing capabilities, the ESR provides IPsec connection capabilities for VPN enabled clients connecting through the ESR.

The ESR is a PCI-104 router module solution for protecting the network. The ESR provides routing, firewall, and VPN Gateway capabilities. The ESR controls the flow of IP traffic by matching information contained in the headers of connection-oriented or connection-less IP packets against a set of rules specified by the Authorized Administrator for firewalls. This header information includes source and destination host (IP) addresses, source and destination port numbers, and the transport service application protocol (TSAP) held within the data field of the IP packet. Depending upon the rule and the results of the match, the firewall either passes or drops the packet. The packet will be denied if the security policy is violated.

In addition to IP header information, the TOE mediates information flows on the basis of other information, such as the direction (incoming or outgoing) of the packet on any given firewall network interface. For connection-oriented transport services, the firewall either permits connections and subsequent packets for the connection or denies the connection and subsequent packets associated with the connection.

The ESR can also establish trusted paths of peer-to-peer VPN tunnels. In addition, the ESR can act as a VPN Gateway by establishing secure VPN tunnels with IPsec VPN clients. Remote VPN clients are able to securely connect into the ESR over an encrypted session in order to connect to an authorized internal private network.

1.2.2 Supported non-TOE / Software/ Firmware

The TOE supports (in some cases optionally) the following hardware, software, and firmware in its environment when the TOE is configured in its evaluated configuration:

Table 3 IT Environment Components

Component	Required	Usage/Purpose Description for TOE performance
Certification Authority	No	This includes any IT Environment Certification Authority on the TOE network. This can be used to provide the TOE with a valid certificate during certificate enrollment.
Chassis	Yes	The router supports Input/Ouput connectors through standard RJ-45 connectors, or any other cPCI compatible network connector. The chassis can be any off-the-shelf module that is capable of holding a PCI-104 form factor.
Local Console	No	This includes any IT Environment Console that is directly connected to the TOE via the Serial Console Port and is used by the TOE administrator to support TOE administration.
Management Workstation with SSH Client	Yes	This includes any IT Environment Management workstation with a SSH client installed that is used by the TOE administrator to support TOE administration through SSH protected channels. Any SSH client that supports SSHv2 may be used.
NTP Server	No	The TOE supports communications with an NTP server.
RADIUS or TACACS+ AAA Server	No	This includes any IT environment RADIUS or TACACS+ AAA server that provides single-use authentication mechanisms. The TOE correctly leverages the services provided by this RADIUS or TACACS+ AAA server to provide single-use authentication to administrators.

Component	Required	Usage/Purpose Description for TOE performance
Remote VPN Endpoint	Yes	This includes any VPN peer or client with which the TOE participates in VPN communications. Remote VPN Endpoints may be any device or software client that supports IPsec VPN communications. Both VPN clients and VPN gateways are considered to be Remote VPN Endpoints by the TOE.
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 v3 communications. Both VPN clients and VPN gateways are considered to be VPN peers by the TOE.
Syslog Server	Yes	The syslog audit server is used for remote storage of audit records that have been generated by and transmitted from the TOE.
USB token	No	The TOE supports the optional storing of digital certificates and private keys on a USB token. A USB token is a smart card with a USB interface. The token can securely store any type of file within its available storage space (32 KB). Configuration files that are stored on the USB token can be encrypted and accessed only via a user PIN. The device does not load the configuration file unless the proper PIN has been configured for secure deployment of device configuration files.

1.3 TOE DESCRIPTION

This section provides an overview of the Cisco X-ES Xpedite5205 Embedded Services Router Target of Evaluation (TOE). The TOE is comprised of both software and hardware. The hardware is comprised of the following: X-ES XPedite5205. The X-ES XPedite5205 is a rebranded Cisco ESR 5930. The software is comprised of the Universal Cisco Internet Operating System (IOS) software image Release IOS 15.2(4)GC.

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

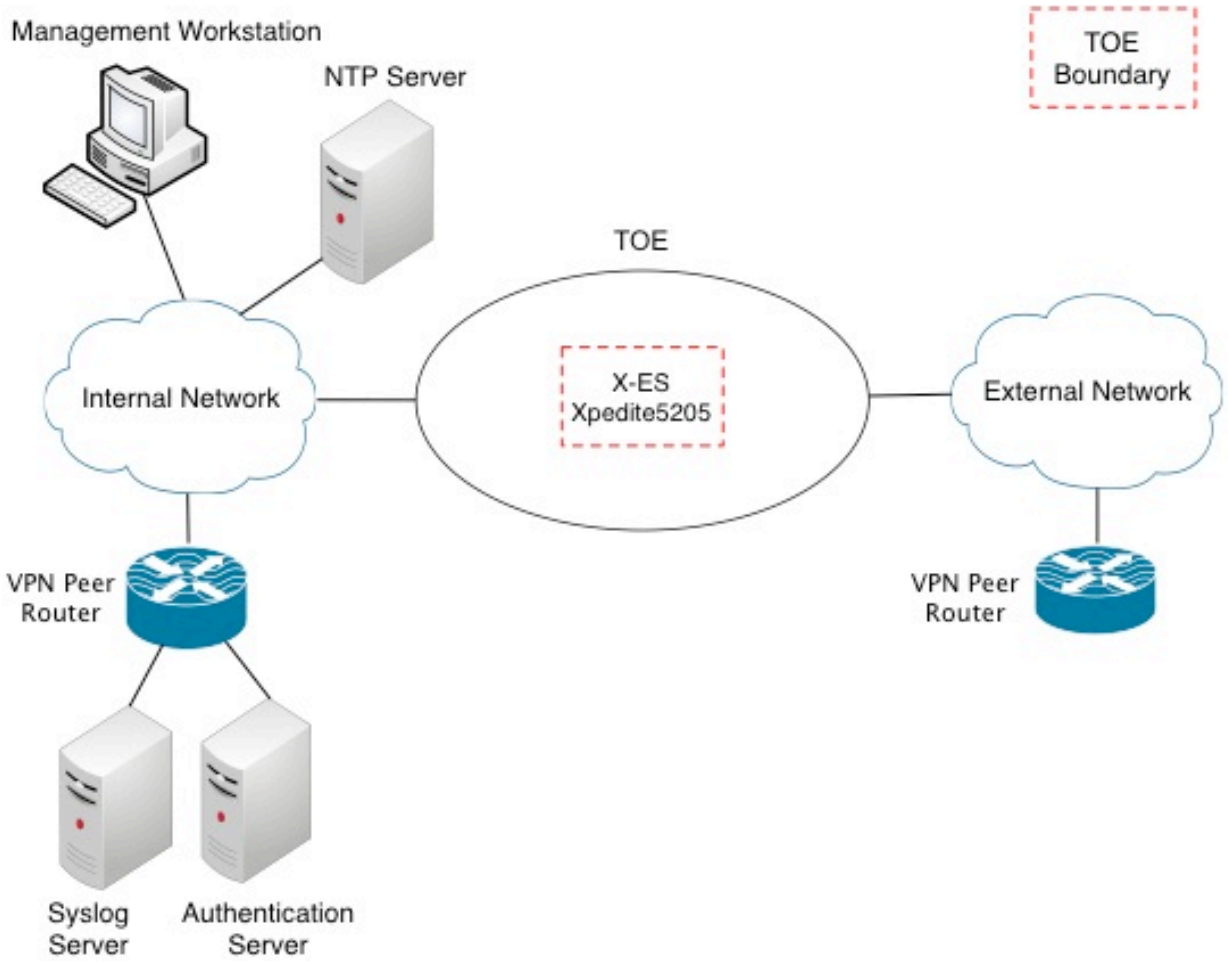

Figure 1 TOE Example Deployment

The previous figure includes the following:

- Example of ESR model:
  - X-ES Xpedite5205
- The following are considered to be in the IT Environment:
  - (2) VPN Peers
  - Management Workstation
  - Authentication Server
  - NTP Server
  - Syslog Server

1.4 TOE Evaluated Configuration

Table 4 Evaluated Configurations

TOE	X-ES Xpedite5205 running IOS 15.2(4) GC (FIPS validated)

The TOE consists of one or more physical devices as specified in section 1.5 below and includes the Cisco IOS software. The TOE has two or more network interfaces and is connected to at least one internal and one external network. The Cisco IOS configuration determines how packets are handled to and from the TOE's network interfaces. The router configuration will determine how traffic flows received on an interface will be handled. Typically, packet flows are passed through the internetworking device and forwarded to their configured destination.

The TOE can optionally connect to an NTP server on its internal network for time services. A syslog server is also used for remote backup of the audit records. If these servers are used, they must be attached to the internal (trusted) network. The internal (trusted) network is meant to be separated effectively from unauthorized individuals and user traffic; one that is in a controlled environment where implementation of security policies can be enforced.

If the TOE is to be remotely administered, SSHv2 must be used for that purpose. All administrative capabilities can be performed either remotely via SSHv2 or locally using the console port. Both methods access the same Command Line Interface (CLI) functionality.

1.5 Physical Scope of the TOE

The TOE is a hardware and software solution that makes up the router models as follows: X-ES XPedite5205. The network, on which they reside, is considered part of the environment. The TOE is comprised of the following physical specifications as described in Table 5 below:

Table 5 Hardware Models and Specifications

Hardware	Interfaces
X-ES XPedite5205 (Standard Air-Cooled)	(1) Serial Console Port (1) Auxiliary Port (4) 10/100/1000 Port LED signals
X-ES XPedite5205 (Rugged Air-Cooled)	(1) Serial Console Port (1) Auxiliary Port (4) 10/100/1000 Port LED signals
X-ES XPedite5205 (Conduction-Cooled)	(1) Serial Console Port (1) Auxiliary Port (4) 10/100/1000 Port LED signals

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. Security Audit
2. Cryptographic Support
3. Full Residual Information Protection
4. Identification and Authentication
5. Security Management
6. Packet Filtering
7. Protection of the TSF
8. TOE Access
9. Trusted Path/Channels

These features are described in more detail in the subsections below. In addition, the TOE implements all RFCs of the NDPPv1.1 and VPN EPv1.1 as necessary to satisfy testing/assurance measures prescribed therein.

1.6.1 Security audit

The Cisco X-ES Xpedite5205 Embedded Services Router provide extensive auditing capabilities. The TOE can audit events related to cryptographic functionality, identification and authentication, and administrative actions. The Cisco ESR routers generate an audit record for each auditable event. For each event, the TOE records the date and time of each event, the type of event, the subject identity, and the outcome of the event. The administrator configures auditable events, backs-up, and manages audit data storage. The TOE provides the audit trail protection by providing remote backup to a syslog server over an encrypted channel. The audit messages include a timestamp that can be provided by the TOE or an optional NTP server in the operational environment.

1.6.2 Cryptographic support

The TOE provides cryptography in support of other Cisco ESR security functionality. This cryptography has been validated for conformance to the requirements of FIPS 140-2 Level 2 (Certificate #2242). Please see Table 6 below for references to the algorithm certificates.

Table 6 FIPS references

Algorithm	Supported Mode	Algorithm Certificate Number	
		IOS	Freescale MPC8548E
AES	CBC (128, 192, 256) CTR (256) GCM (128, 192, 256)	2784	962, 1535
Triple-DES	KO 1 & 2, CBC	1672	757

Algorithm	Supported Mode	Algorithm Certificate Number	
		IOS	Freescale MPC8548E
SHS (SHA-1, 256, and 512)	Byte Oriented	2339	933
HMAC SHA-1	Byte Oriented	1743	537
DRBG	CTR (using AES-256)	471	N/A
RSA	2048-3072 bit key	1456	N/A
ECDSA	P-256, P-384	485	N/A

The TOE provides cryptography in support of VPN connections and remote administrative management via SSHv2. The cryptographic services provided by the TOE are described in Table 6 below.

Table 7 TOE Provided Cryptography

Cryptographic Method	Use within the TOE
Internet Key Exchange	Used to establish initial IPsec session.
Secure Shell Establishment	Used to establish initial SSH session.
RSA Signature Services	Used in IPsec session establishment. Used in SSH session establishment.
SP 800-90 RBG	Used in IPsec session establishment. Used in SSH session establishment.
SHS	Used to provide IPsec traffic integrity verification Used to provide SSH traffic integrity verification
AES	Used to encrypt IPsec session traffic. Used to encrypt SSH session traffic.

The TOE can act as a certification authority thus signing and issuing certificates to the TOE and other devices. The TOE can also use the X.509v3 certificate for securing IPsec and SSH sessions.

1.6.3 Full residual information protection

The TOE ensures that all information flows from the TOE do not contain residual information from previous traffic. Packets are padded with zeros. Residual data is never transmitted from the TOE.

1.6.4 Identification and authentication

The TOE performs two types of authentication: device-level authentication of the remote device (VPN peers) and user authentication for the Authorized Administrator of the TOE. Device-level authentication allows the TOE to establish a secure channel with a trusted peer. The secure channel is established only after each device authenticates the other. Device-level authentication is performed via IKE/IPsec mutual authentication. The IKE phase authentication for the IPsec communication channel between the TOE and authentication server and between the TOE and

syslog server is considered part of the Identification and Authentication security functionality of the TOE.

The TOE provides authentication services for administrative users to connect to the TOEs secure CLI administrator interface. The TOE requires Authorized Administrators to authenticate prior to being granted access to any of the management functionality. The TOE can be configured to require a minimum password length of 15 characters as well as mandatory password complexity rules that includes special characters. The TOE provides administrator authentication against a local user database. Password-based authentication can be performed on the serial console or SSH interfaces. The SSHv2 interface also supports authentication using SSH keys. The TOE optionally supports use of a RADIUS or TACACS+ AAA server (part of the IT Environment) to facilitate authentication (including single-use authentication, or password-based authentication) for administrative users attempting to connect to the TOE's CLI.

The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for IPsec and SSH connections.

1.6.5 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 or via a local console connection. The TOE provides the ability to securely manage:

- All TOE administrative users;
- All identification and authentication;
- All audit functionality of the TOE
- All TOE cryptographic functionality;
- The timestamps maintained by the TOE;
- Update to the TOE; and
- TOE configuration file storage and retrieval.

The TOE supports two separate administrative roles: non-privileged Administrator and privileged Administrator. Only the privileged administrator can perform all of the above security relevant management functions. The privileged Administrator is also considered to be the Authorized Administrator and Security Administrator.

Administrators can create configurable login banners to be displayed at time of login, and can also define an inactivity timeout for each admin interface to terminate sessions after a set period of inactivity.

1.6.6 Packet Filtering

The TOE provides packet filtering and secure IPsec tunneling. The tunnels can be established between two trusted VPN peers as well as between remote VPN clients and the TOE. More accurately, these tunnels are sets of security associations (SAs). The SAs define the protocols

and algorithms to be applied to sensitive packets and specify the keying material to be used. SAs are unidirectional and are established per the ESP security protocol. An authorized administrator can define the traffic that needs to be protected via IPsec by configuring access lists (permit, deny, log) and applying these access lists to interfaces using crypto map sets.

1.6.7 Protection of the TSF

The TOE protects against interference and tampering by untrusted subjects by implementing identification, authentication, and access controls to limit configuration to Authorized Administrators. The TOE prevents reading of cryptographic keys and passwords. Additionally Cisco IOS is not a general-purpose operating system and access to Cisco IOS memory space is restricted to only Cisco IOS functions.

Use of separate VLANs is used to ensure routing protocol communications between the TOE and neighbor routers including routing table updates and neighbor router authentication will be logically isolated from traffic on other VLANs.

The TOE internally maintains the date and time. This date and time is used as the timestamp that is applied to audit records generated by the TOE. Administrators can update the TOE's clock manually, or can configure the TOE to use NTP to synchronize the TOE's clock with an external time source. Finally, the TOE performs testing to verify correct operation of the router itself and that of the cryptographic module. Whenever any system failures occur within the TOE the TOE will cease operation.

The TOE is able to verify any software updates prior to the software updates being installed on the TOE to avoid the installation of malicious software.

1.6.8 TOE Access

The TOE can terminate inactive sessions after an Authorized Administrator configurable time-period. Once a session has been terminated, the TOE requires the user to re-authenticate to establish a new session.

The TOE can also display an Authorized Administrator specified banner on the CLI management interface prior to allowing any administrative access to the TOE.

1.6.9 Trusted Path/Channels

The TOE allows trusted paths to be established to itself from remote administrators over SSHv2, and initiates outbound IPsec tunnels to transmit audit messages to remote syslog servers. In addition, IPsec is used to secure the session between the TOE and the authentication servers. The TOE can also establish trusted paths of peer-to-peer VPN tunnels. The peer-to-peer VPN tunnels can be used for securing the session between the TOE and authentication server/syslog server. In addition, the TOE can establish secure VPN tunnels with IPsec VPN clients. Remote

VPN clients are able to securely connect into the X-ES over an encrypted session in order to connect to an authorized internal private network.

The TOE also supports direct connections from VPN clients, and protects against threats related to those client connections. The TOE disconnects sessions that have been idle too long, can be configured to deny sessions based on IP, time, and day, and can be configured to NAT external IPs of connecting VPN clients to internal network addresses.

1.7 Excluded Functionality

The following functionality is excluded from the evaluation.

Table 8 Excluded Functionality

Excluded Functionality	Exclusion Rationale
Non-FIPS 140-2 mode of operation on the TOE	This mode of operation allows cryptographic operations that are not FIPS-approved.

These services will be disabled by configuration. The exclusion of this functionality does not affect compliance to the NDPP and VPNEP.

2 CONFORMANCE CLAIMS

2.1 Common Criteria Conformance Claim

The TOE and ST are compliant with the Common Criteria (CC) Version 3.1, Revision 4, dated: September 2012.

The TOE and ST are CC Part 2 extended and CC Part 3 conformant.

2.2 Protection Profile Conformance

The TOE and ST are conformant with the Protection Profile for Network Devices (NDPP).

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

Table 9 Protection Profiles

Protection Profile	Version	Date
Protection Profile for Network Devices (NDPP)	1.1	June 8, 2012
Security Requirements for Network Devices Errata #2		13 January 2013
Network Device Protection Profile Extended Package VPN Gateway (VPNEP)	1.1	April 12, 2013

2.3 Protection Profile Conformance Claim Rationale

2.3.1 TOE Appropriateness

The TOE provides all of the functionality at a level of security commensurate with that identified in the U.S. Government Protection Profile:

- Protection Profile for Network Devices v1.1
- Network Device Protection Profile Extended Package VPN Gateway v1.1

2.3.2 TOE Security Problem Definition Consistency

The Assumptions, Threats, and Organization Security Policies included in the Security Target represent the Assumptions, Threats, and Organization Security Policies specified in the NDPPv1.1 and VPN EPv1.1 for which conformance is claimed verbatim. All concepts covered in the Protection Profile Security Problem Definition are included in the Security Target Statement of Security Objectives Consistency.

The Security Objectives included in the Security Target represent the Security Objectives specified in the NDPPv1.1 and VPN EPv1.1 for which conformance is claimed verbatim. All concepts covered in the Protection Profile's Statement of Security Objectives are included in the Security Target.

2.3.3 Statement of Security Requirements Consistency

The Security Functional Requirements included in the Security Target represent the Security Functional Requirements specified in the NDPPv1.1 and VPN EPv1.1 for which conformance is claimed verbatim. All concepts covered in the Protection Profile's Statement of Security Requirements are included in this Security Target. Additionally, the Security Assurance Requirements included in this Security Target are identical to the Security Assurance Requirements included in section 4.3 of the NDPPv1.1 and section 5.2 of the VPN EPv1.1.

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.

3.1 Assumptions

The specific conditions listed in the following subsections are assumed to exist in the TOE's 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 10 TOE Assumptions

Assumption	Assumption Definition
Reproduced from the NDPPv1.1	
A.NO_GENERAL_PURPOSE	It is assumed that there are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of 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.TRUSTED_ADMIN	TOE Administrators are trusted to follow and apply all administrator guidance in a trusted manner.
Reproduced from the VPNEPv1.1	
A.CONNECTIONS	It is assumed that the TOE is connected to distinct networks in a manner that ensures that the TOE security policies will be enforced on all applicable network traffic flowing among the attached networks.

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 Enhanced-Basic.

Table 11 Threats

Threat	Threat Definition
Reproduced from the NDPPv1.1	
T.ADMIN_ERROR	An administrator may unintentionally install or configure the TOE incorrectly, resulting in ineffective security mechanisms.
T.TSF_FAILURE	Security mechanisms of the TOE may fail, leading to a compromise of the TSF.
T.UNDETECTED_ACTIONS	Malicious remote users or external IT entities may take actions that adversely affect the security of the TOE. These actions may remain undetected and thus their effects cannot be effectively mitigated.

Threat	Threat Definition
T.UNAUTHORIZED_ACCESS	A user may gain unauthorized access to the TOE data and TOE executable code. A malicious user, process, or external IT entity may masquerade as an authorized entity in order to gain unauthorized access to data or TOE resources. A malicious user, process, or external IT entity may misrepresent itself as the TOE to obtain identification and authentication data.
T.UNAUTHORIZED_UPDATE	A malicious party attempts to supply the end user with an update to the product that may compromise the security features of the TOE.
T.USER_DATA_REUSE	User data may be inadvertently sent to a destination not intended by the original sender.
Reproduced from the VPNEPv1.1	
T.NETWORK_DISCLOSURE	Sensitive information on a protected network might be disclosed resulting from ingress- or egress-based actions.
T. NETWORK_ACCESS	Unauthorized access may be achieved to services on a protected network from outside that network, or alternately services outside a protected network from inside the protected network.
T.NETWORK_MISUSE	Access to services made available by a protected network might be used counter to Operational Environment policies.
T.TSF_FAILURE	Security mechanisms of the TOE mail fail, leading to a compromise of the TSF.
T.REPLAY_ATTACK	If malicious or external IT entities are able to gain access to the network, they may have the ability to capture information traversing throughout the network and send them on to the intended receiver.
T.DATA_INTEGRITY	A malicious party attempts to change the data being sent – resulting in loss of integrity.

3.3 Organizational Security Policies

The following table lists the Organizational Security Policies imposed by an organization to address its security needs.

Table 12 Organizational Security Policies

Policy Name	Policy Definition
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.

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 13 Security Objectives for the TOE

TOE Objective	TOE Security Objective Definition
Reproduced from the NDPPv1.1	
O.PROTECTED_COMMUNICATIONS	The TOE will provide protected communication channels for administrators, other parts of a distributed TOE, and authorized IT entities.
O.VERIFIABLE_UPDATES	The TOE will provide the capability to help ensure that any updates to the TOE can be verified by the administrator to be unaltered and (optionally) from a trusted source.
O.SYSTEM_MONITORING	The TOE will provide the capability to generate audit data and send those data to an external IT entity.
O.DISPLAY_BANNER	The TOE will display an advisory warning regarding use of the TOE.
O.TOE_ADMINISTRATION	The TOE will provide mechanisms to ensure that only administrators are able to log in and configure the TOE, and provide protections for logged-in administrators.
O.RESIDUAL_INFORMATION_CLEARING	The TOE will ensure that any data contained in a protected resource is not available when the resource is reallocated.
O.SESSION_LOCK	The TOE shall provide mechanisms that mitigate the risk of unattended sessions being hijacked.
O.TSF_SELF_TEST	The TOE will provide the capability to test some subset of its security functionality to ensure it is operating properly.
Reproduced from the VPNEPv1.1	
O.ADDRESS_FILTERING	The TOE will provide the means to filter and log network packets based on source and destination addresses.
O.AUTHENTICATION	The TOE will provide a means to authenticate the user to ensure they are communicating with an authorized external IT entity.
O.CRYPTOGRAPHIC_FUNCTIONS	The TOE will provide means to encrypt and decrypt data as a means to maintain confidentiality and allow for detection and modification of TSF data that is transmitted outside of the TOE
O.FAIL_SECURE	Upon a self-test failure, the TOE will shutdown to ensure data cannot be passed while not adhering to the security policies configured by the administrator.

TOE Objective	TOE Security Objective Definition
O.PORT_FILTERING	The TOE will provide the means to filter and log network packets based on source and destination transport layer ports.

4.2 Security Objectives for the Environment

All of the assumptions stated in section 3.1 are considered to be security objectives for the environment. The following are the Protection Profile non-IT security objectives, which, in addition to those assumptions, are to be satisfied without imposing technical requirements on the TOE. That is, they will not require the implementation of functions in the TOE hardware and/or software. Thus, they will be satisfied largely through application of procedural or administrative measures.

Table 14 Security Objectives for the Environment

Environment Security Objective	IT Environment Security Objective Definition
Reproduced from the NDPPv1.1	
OE.NO_GENERAL_PURPOSE	There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE.
OE.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.
OE.TRUSTED_ADMIN	TOE Administrators are trusted to follow and apply all administrator guidance in a trusted manner.
Reproduced from the VPNEPv1.1	
OE.CONNECTIONS	TOE administrators will ensure that the TOE is installed in a manner that will allow the TOE to effectively enforce its policies on network traffic flowing among attached networks.

5 SECURITY REQUIREMENTS

This section identifies the Security Functional Requirements for the TOE. The Security Functional Requirements included in this section are derived from Part 2 of the *Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 4, dated: September 2012* and all 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:

- Where operations were completed in the NDPP itself, the formatting used in the NDPP has been retained;
- Assignment: Indicated with *italicized* text, which may or may not be bracketed;
- Refinement made by PP author: Indicated with **bold** text; may have **Refinement:** at the beginning of the element for further clarification.
- Selection: Indicated with underlined text, which may or may not be bracketed;
- 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 15 Security Functional Requirements

Class Name	Component Identification	Component Name
Reproduced from the Protection Profile for Network Devices and VPN EP		
FAU: Security audit	FAU_GEN.1	Audit data generation
	FAU_GEN.2	User Identity Association
	FAU_STG_EXT.1	External Audit Trail Storage
FCS: Cryptographic support	FCS_CKM.1	Cryptographic Key Generation (for asymmetric keys)
	FCS_CKM.1(2)	Cryptographic Key Generation (for asymmetric keys)
	FCS_CKM_EXT.4	Cryptographic Key Zeroization
	FCS_COP.1(1)	Cryptographic Operation (for data encryption/decryption)
	FCS_COP.1(2)	Cryptographic Operation (for cryptographic signature)
	FCS_COP.1(3)	Cryptographic Operation (for cryptographic hashing)
	FCS_COP.1(4)	Cryptographic Operation (for keyed-hash message authentication)
	FCS_IPSEC_EXT.1	Explicit: IPSEC
	FCS_SSH_EXT.1	Explicit: SSH
	FCS_RBG_EXT.1	Extended: Cryptographic Operation (Random Bit Generation)
FDP: User data protection	FDP_RIP.2	Full Residual Information Protection
FIA: Identification and	FIA_AFL.1	Authentication Failure Handling

Class Name	Component Identification	Component Name
authentication	FIA_PMG_EXT.1	Password Management
	FIA_UIA_EXT.1	User Identification and Authentication
	FIA_UAU_EXT.2	Extended: Password-based Authentication Mechanism
	FIA_UAU.7	Protected Authentication Feedback
	FIA_X509_EXT.1	FIA_X509_EXT.1 Extended: X.509 Certificates
	FMT_MOF.1	Management of Security Functions Behavior
FMT: Security management	FMT_MTD.1	Management of 7TSF Data (for general TSF data)
	FMT_SMF.1	Specification of Management Functions
	FMT_SMR.2	Restrictions on Security Roles
FPF: Packet Filtering	FPF_RUL_EXT.1	Packet Filtering
FPT: Protection of the TSF	FPT_FLS.1	Fail Secure
	FPT_SKP_EXT.1	Extended: Protection of TSF Data (for reading of all symmetric keys)
	FPT_APW_EXT.1	Extended: Protection of Administrator Passwords
	FPT_STM.1	Reliable Time Stamps
	FPT_TUD_EXT.1	Extended: Trusted Update
	FPT_TST_EXT.1	TSF Testing
FTA: TOE Access	FTA_SSL_EXT.1	TSF-initiated Session Locking
	FTA_SSL.3	TSF-initiated Termination
	FTA_SSL.4	User-initiated Termination
	FTA_TAB.1	Default TOE Access Banners
FTP: Trusted path/channels	FTP_ITC.1	Trusted Channel
	FTP_TRP.1	Trusted Path

5.3 SFRs Drawn from NDPP and VPN GW EP

5.3.1 Security audit (FAU)

5.3.1.1 FAU_GEN.1 Audit data generation

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

a) Start-up and shutdown of the audit functions;
b) All auditable events for the not specified level of audit; and
c) All administrative actions;
d) [Specifically defined auditable events listed in Table 15

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

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

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* Table 15].

Table 16 Auditable Events

SFR	Auditable Event	Additional Audit Record Contents
Audit Events and Details from NDPP		
FAU_GEN.1	None.	None.
FAU_GEN.2	None.	None.
FAU_STG_EXT.1	None.	None.
FCS_CKM.1	None.	None.
FCS_CKM_EXT.4	None.	None.
FCS_COP.1(1)	None.	None.
FCS_COP.1(2)	None.	None.
FCS_COP.1(3)	None.	None.
FCS_COP.1(4)	None.	None.
FCS_IPSEC_EXT.1	Failure to establish an IPsec SA. Establishment/Termination of an IPsec SA.	Reason for failure. Non-TOE endpoint of connection (IP address) for both successes and failures.
FC_SSH_EXT.1	Failure to establish an SSH session Establishment/Termination of an SSH session.	Reason for failure. Non-TOE endpoint of connection (IP address) for both successes and failures.
FCS_RBG_EXT.1	None.	None.
FDP_RIP.2	None.	None.
FIA_PMG_EXT.1	None.	None.
FIA_UIA_EXT.1	All use of the identification and authentication mechanism.	Provided user identity, origin of the attempt (e.g., IP address).
FIA_UAU_EXT.2	All use of the authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UAU.7	None.	None.
FMT_MTD.1	None.	None.
FMT_SMF.1	None.	None.
FMT_SMR.2	None.	None.
FPT_SKP_EXT.1	None.	None.
FPT_APW_EXT.1	None.	None.
FPT_STM.1	Changes to the time.	The old and new values for the time. Origin of the attempt (e.g., IP address).
FPT_TUD_EXT.1	Initiation of update.	No additional information.
FPT_TST_EXT.1	None.	None.
FTA_SSL_EXT.1	Any attempts at unlocking of an interactive session.	No additional information.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	No additional information.
FTA_SSL.4	The termination of an interactive session.	No additional information.
FTA_TAB.1	None.	None.
FTP_ITC.1	Initiation of the trusted channel. Termination of the trusted channel. Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempt.
FTP_TRP.1	Initiation of the trusted channel. Termination of the trusted channel. Failures of the trusted path functions.	Identification of the claimed user identity.
Audit Events and Details from VPN EP		
FCS_IPSEC_EXT.1	Session Establishment with peer	Source and destination addresses Source and destination ports TOE Interface
FIA_X509_EXT.1	Establishing session with CA	Source and destination addresses Source and destination ports

SFR	Auditable Event	Additional Audit Record Contents
		TOE Interface
FPF_RUL_EXT.1	Application of rules configured with the 'log' operation	Source and destination addresses Source and destination ports Transport Layer Protocol TOE Interface
	Indication of packets dropped due to too much network traffic	TOE interface that is unable to process packets

5.3.1.2 FAU_GEN.2 User Identity Association

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

5.3.1.3 FAU_STG_EXT.1 External Audit Trail Storage

FAU_STG_EXT.1.1 The TSF shall be able to transmit the generated audit data to an external IT entity using a trusted channel implementing the IPsec protocol.

5.3.2 Cryptographic Support (FCS)

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

FCS_CKM.1.1 Refinement: The TSF shall generate **asymmetric** cryptographic keys **used for key establishment** in accordance with:

- *NIST Special Publication 800-56A, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" for elliptic curve-based key establishment schemes and implementing "NIST curves" P-256, P-384 and* P-521] (as defined in FIPS PUB 186-3, "Digital Signature Standard")
- *NIST Special Publication 800-56A, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" for finite field-based key establishment schemes;*
- [NIST Special Publication 800-56B, "Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography" for RSA-based key establishment schemes]

and specified cryptographic key sizes equivalent to, or greater than, a symmetric key strength of 112 bits.

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

FCS_CKM.1.2 Refinement: The TSF shall generate **asymmetric** cryptographic keys **used for IKE peer authentication** in accordance with a:

[

- *FIPS PUB 186-3, "Digital Signature Standard (DSS)", Appendix B.3 for RSA schemes;*
- *FIPS PUB 186-3, "Digital Signature Standard (DSS)", Appendix B.4 for ECDSA schemes and implementing "NIST curves" P-256, P-384 and* [P-521]*;*
- *ANSI X9.31-1998, Appendix A.2.4 Using AES for RSA schemes*]

and specified cryptographic key sizes *equivalent to, or greater than, a symmetric key strength of 112 bits.*

5.3.2.3 FCS_CKM_EXT.4 Cryptographic Key Zeroization

FCS_CKM_EXT.4.1 The TSF shall zeroize all plaintext secret and private cryptographic keys and CSPs when no longer required.

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

FCS_COP.1.1(1) Refinement: The TSF shall perform [*encryption and decryption*] in accordance with a specified cryptographic algorithm *AES operating in* **GCM, CBC,** [*no other modes*] and cryptographic key sizes 128-bits, 256-bits, and [**192 bits**] that meets the following:

- **FIPS PUB 197, "Advanced Encryption Standard (AES)"**
- **[NIST SP 800-38D, NIST SP 800-38A [no other standards]**

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

FCS_COP.1.1(2) Refinement: The TSF shall perform **cryptographic signature services** in accordance with a :

- **RSA Digital Signature Algorithm (RSA) with a key size (modulus) of 2048 bits or greater that meets FIPS PUB 186-2 or FIPS PUB 186-3, "Digital Signature Standard",**
- **Elliptic Curve Digital Signature Algorithm (ECDSA) with a key size of 256 bits or greater that meets FIPS PUB 186-3, "Digital Signature Standard" with "NIST curves" P-256, P-384 and [P-521] (as defined in FIPS PUB 186-3, "Digital Signature Standard").**

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

FCS_COP.1.1(3) Refinement: The TSF shall perform [*cryptographic hashing services*] in accordance with a specified cryptographic algorithm **SHA-1, SHA-256, SHA-384, SHA-512 and message digest sizes 160, 256, 384, 512 bits** that meet the following: *FIPS Pub 180-3, "Secure Hash Standard."*

5.3.2.7 FCS_COP.1(4) Cryptographic Operation (for keyed-hash message authentication)

FCS_COP.1.1(4) Refinement: The TSF shall perform [*keyed-hash message authentication*] in accordance with a specified cryptographic algorithm HMAC-**[SHA-1], key size [*160- bits*], and message digest sizes [160] bits** that meet the following: [*FIPS Pub 198-1, "The Keyed-Hash Message Authentication Code, and FIPS Pub 180-3, "Secure Hash Standard."*]

5.3.2.8 FCS_IPSEC_EXT.1 Explicit: IPSEC

FCS_IPSEC_EXT.1.1 The TSF shall implement the IPsec protocol ESP as defined by RFC 4301.

FCS_IPSEC_EXT.1.2 The TSF shall implement [tunnel mode, transport mode].

FCS_IPSEC_EXT.1.3 The TSF shall have a nominal, final entry in the SPD that matches anything that is otherwise unmatched, and discards it.

FCS_IPSEC_EXT.1.4 The TSF shall implement the IPsec protocol ESP as defined by RFC 4303 using the cryptographic algorithms AES-GCM-128, AES-GCM-256 as specified in RFC 4106, [AES-CBC-128, AES-CBC-256 (both specified by RFC 3602) together with a Secure Hash Algorithm (SHA)-based HMAC].
f
FCS_IPSEC_EXT.1.5 The TSF shall implement the protocol: [IKEv1 as defined in RFCs 2407, 2408, 2409, RFC 4109, [no other RFCs for extended sequence numbers] and [no other RFCs for hash functions]; IKEv2 as defined in RFCs 5996 (with mandatory support for NAT traversal as specified in section 2.23) and [no other RFCs for hash functions]].

FCS_IPSEC_EXT.1.6 The TSF shall ensure the encrypted payload in the [IKEv1, IKEv2] protocol uses the cryptographic algorithms AES-CBC-128, AES-CBC-256 as specified in RFC 6379 and [AES-GCM-128, AES-GCM-256 as specified in RFC 5282].

FCS_IPSEC_EXT.1.7 The TSF shall ensure that IKEv1 Phase 1 exchanges use only main mode.

FCS_IPSEC_EXT.1.8 The TSF shall ensure that [IKEv2 SA lifetimes can be configured by an Administrator based on number of packets or length of time, where the time values can be limited to: 24 hours for Phase 1 SAs and 8 hours for Phase 2 SAs, IKEv1 SA lifetimes can be configured by an Administrator based on number of packets or length of time, where the time values can be limited to: 24 hours for Phase 1 SAs and 8 hours for Phase 2 SAs].

FCS_IPSEC_EXT.1.9 The TSF shall generate the secret value x used in the IKE Diffie-Hellman key exchange ("x" in g^x mod p) using the random bit generator specified in FCS_RBG_EXT.1, and having a length of at least [*320 (for DH Group 14), 256 (for DH Group 19), 256 (for DH Group 24), 384 (for DH Group 20), 424 (for DH Group 15), and 480 (bits for DH Group 16)*] bits.

FCS_IPSEC_EXT.1.10 The TSF shall generate nonces used in IKE exchanges 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^[*128*].

FCS_IPSEC_EXT.1.11 The TSF shall ensure that all IKE protocols implement DH Groups 14 (2048-bit MODP), 19 (256-bit Random ECP), and [24 (2048-bit MODP with 256-bit POS), 20 (384-bit Random ECP), *15 (3072 bit MODP), and 16 (4096-bit MODP)*].

FCS_IPSEC_EXT.1.12 The TSF shall ensure that all IKE protocols perform peer authentication using a [RSA, ECDSA] that use X.509v3 certificates that conform to RFC 4945 and [Pre-shared Keys].

FCS_IPSEC_EXT.1.13 The TSF shall be able to ensure by default that the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the [IKEv1 Phase 1, IKEv2 IKE_SA] connection is greater than or equal to the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the [IKEv1 Phase 2, IKEv2 CHILD_SA] connection.

5.3.2.9 FCS_RBG_(EXT).1 Extended: Cryptographic Operation (Random Bit Generation)

FCS_RBG_EXT.1.1 The TSF shall perform all random bit generation (RBG) services in accordance with [NIST Special Publication 800-90 using CTR_DRBG (AES)] seeded by an entropy source that accumulated entropy from a TSF-hardware based noise source, and [no other noise source].

FCS_RBG_EXT.1.2 The deterministic RBG shall be seeded with a minimum of 256 bits of entropy at least equal to the greatest security strength of the keys and hashes that it will generate.

5.3.2.10 FCS_SSH_EXT.1 Explicit: SSH

FCS_SSH_EXT.1.1 The TSF shall implement the SSH protocol that complies with RFCs 4251, 4252, 4253, 4254, and [no other RFCs].

FCS_SSH_EXT.1.2 The TSF shall ensure that the SSH protocol implementation supports the following authentication methods as described in RFC 4252: public key-based, password-based.

FCS_SSH_EXT.1.3 The TSF shall ensure that, as described in RFC 4253, packets greater than [*35,000*] bytes in an SSH transport connection are dropped.

FCS_SSH_EXT.1.4 The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms: AES-CBC-128, AES-CBC-256, *no other algorithms*.

FCS_SSH_EXT.1.5 The TSF shall ensure that the SSH transport implementation uses SSH_RSA and *no other public key algorithms* as its public key algorithm(s).

FCS_SSH_EXT.1.6 The TSF shall ensure that data integrity algorithms used in SSH transport connection is *hmac-sha1*, *hmac-sha1-96*.

FCS_SSH_EXT.1.7 The TSF shall ensure that diffie-hellman-group14-sha1 and no other methods are the only allowed key exchange method used for the SSH protocol.

5.3.3 User data protection (FDP)

5.3.3.1 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 allocation of the resource to all objects.

5.3.4 Identification and authentication (FIA)

5.3.4.1 FIA_AFL.1 Authentication Failure Handling

FIA_AFL.1.1 Refinement: The TSF shall detect when **an Administrator configurable positive integer of successive** unsuccessful authentication attempts occur related to **administrators attempting to authenticate remotely**.

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been **met**, the TSF shall [prevent the offending remote administrator from successfully authenticating until [*an authorized administrator unlocks the locked user account*] is taken by a local Administrator.

5.3.4.2 FIA_PMG_EXT.1 Password Management

FIA_PMG_EXT.1.1 The TSF shall provide the following password management capabilities for administrative passwords:

1. Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters: "!", "@", "#", "$", "%", "^", "&", "*", "(", ")", [*no other characters*];
2. Minimum password length shall settable by the Security Administrator, and support passwords of 15 characters or greater;

5.3.4.3 FIA_PSK_EXT.1 Extended: Pre-Shared Key Composition

FIA_PSK_EXT.1.1 The TSF shall be able to use pre-shared keys for IPsec and [no other protocols].

FIA_PSK_EXT.1.2 The TSF shall be able to accept text-based pre-shared keys that:

- are 22 characters and [*any combination of alphanumeric or special characters up to 128 bytes*];
- composed of any combination of upper and lower case letters, numbers, and

special characters (that include: “!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, and “)”).

FIA_PSK_EXT.1.3 The TSF shall condition the text-based pre-shared keys by using [SHA-1 *or AES*].

FIA_PSK_EXT.1.4 The TSF shall be able to [accept] bit-based pre-shared keys.

5.3.4.4 FIA_UIA_EXT.1 User Identification and Authentication

FIA_UIA_EXT.1.1 The TSF shall allow the following actions prior to requiring the non-TOE entity to initiate the identification and authentication process:

- Display the warning banner in accordance with FTA_TAB.1;
- no other actions.

FIA_UIA_EXT.1.2 The TSF shall require each administrative user to be successfully identified and authenticated before allowing any other TSF-mediated action on behalf of that administrative user.

5.3.4.5 FIA_UAU_EXT.2 Extended: Password-based Authentication Mechanism

FIA_UAU_EXT.2.1 The TSF shall provide a local password-based authentication mechanism, [remote password-based authentication via RADIUS and TACACS+] to perform administrative user authentication.

5.3.4.6 FIA_UAU.7 Protected Authentication Feedback

FIA_UAU.7.1 The TSF shall provide only *obscured feedback* to the administrative user while the authentication is in progress at the local console.

5.3.4.7 FIA_X509_EXT.1 Extended: X.509 Certificates

FIA_X509_EXT.1.1 The TSF shall use X.509v3 certificates as defined by RFC 5280 to support authentication for IPsec and [SSH] connections.

FIA_X509_EXT.1.2 The TSF shall store and protect certificate(s) from unauthorized deletion and modification.

FIA_X509_EXT.1.3 The TSF shall provide the capability for authenticated Administrators to load X.509v3 certificates into the TOE for use by the security functions specified in this PP.

FIA_X509_EXT.1.4 The TSF shall generate a Certificate Request Message as specified in RFC 2986 and be able to provide the following information in the request: public key, Common Name, Organization, Organizational Unit, and Country.

FIA_X509_EXT.1.5 The TSF shall validate the certificate using [Online Certificate Status Protocol (OCSP) as specified in RFC 2560, a Certificate Revocation List (CRL) as specified in

RFC 5759].

FIA_X509_EXT.1.6 The TSF shall validate a certificate path by ensuring the presence of the basicConstraints extension is present and the cA flag is set to TRUE for all CA certificates.

FIA_X509_EXT.1.7 The TSF shall not treat a certificate as a CA certificate if the basicConstraints extension is not present or the cA flag is not set to TRUE.

FIA_X509_EXT.1.8 The TSF shall not establish an SA if a certificate is deemed invalid.

FIA_X509_EXT.1.9 The TSF shall not establish an SA if the distinguished name (DN) contained in a certificate does not match the expected DN for the entity attempting to establish a connection.

FIA_X509_EXT.1.10 When the TSF cannot establish a connection to determine the validity of a certificate, the TSF shall, at the option of the administrator, establish an SA or disallow the establishment of an SA.

5.3.5 Security management (FMT)

5.3.5.1 FMT_MOF.1 Management of Security Functions Behavior

FMT_MOF.1.1 Refinement: The TSF shall restrict the ability to enable, disable, determine and modify the behavior of all of the security functions of the TOE identified in this EP to an authenticated Administrator.

5.3.5.2 FMT_MTD.1 Management of TSF Data (for general TSF data)

FMT_MTD.1.1 The TSF shall restrict the ability to *manage* the *TSF data* to the *Security Administrators*.

5.3.5.3 FMT_SMF.1 Specification of Management Functions

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

- *Ability to administer the TOE locally and remotely;*
- *Ability to update the TOE, and to verify the updates using [published hash] capability prior to installing those updates;*
- *Ability to configure the cryptographic functionality;*
- *Ability to configure the IPsec functionality,*
- *Ability to enable, disable, determine and modify the behavior of all the security functions of the TOE identified in this EP to the Administrator,*
- *Ability to configure all security management functions identified in other sections of this EP.*

5.3.5.4 FMT_SMR.2 Restrictions on Security Roles

FMT_SMR.2.1 The TSF shall maintain the roles:
- Authorized Administrator.

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
- Authorized Administrator role shall be able to administer the TOE locally;
- Authorized Administrator role shall be able to administer the TOE remotely;

are satisfied.

5.3.6 Packet Filtering (FPF)

5.3.6.1 FPF_RUL_EXT.1 Packet Filtering

FPF_RUL_EXT.1.1 The TSF shall perform Packet Filtering on network packets processed by the TOE.

FPF_RUL_EXT.1.2 The TSF shall process the following network traffic protocols:
- Internet Protocol (IPv4)
- Internet Protocol version 6 (IPv6)
- Transmission Control Protocol (TCP)
- User Datagram Protocol (UDP)

and be capable of inspecting network packet header fields defined by the following RFCs to the extent mandated in the other elements of this SFR
- RFC 791 (IPv4)
- RFC 2460 (IPv6)
- RFC 793 (TCP)
- RFC 768 (UDP).

FPF_RUL_EXT.1.3 The TSF shall allow the definition of Packet Filtering rules using the following network protocol fields:
- IPv4
  - Source address
  - Destination Address
  - Protocol
- IPv6
  - Source address
  - Destination Address
  - Next Header (Protocol)
- TCP
  - Source Port
  - Destination Port

- UDP
    - Source Port
    - Destination Port

and distinct interface.

Application Note: There was a minor revision to the NDPPv1.1 errata 2 by NIAP that states that the IPv6 extension headers do not have to be tested.

FPF_RUL_EXT.1.4 The TSF shall allow the following operations to be associated with Packet Filtering rules: permit, deny, and log.

FPF_RUL_EXT.1.5 The TSF shall allow the Packet Filtering rules to be assigned to each distinct network interface.

FPF_RUL_EXT.1.5 The TSF shall allow the Packet Filtering rules to be assigned to each distinct network interface.

FPF_RUL_EXT.1.6 The TSF shall process the applicable Packet Filtering rules (as determined in accordance with FPF_RUL_EXT.1.5) in the following order: Administrator-defined.

FPF_RUL_EXT.1.7 The TSF shall deny packet flow if a matching rule is not identified.

5.3.7 Protection of the TSF (FPT)

5.3.7.1 FPT_FLS.1 Fail Secure

FPT_FLS.1.1 Refinement: The TSF shall **shutdown** when the following types of failures occur: failure of the power-on self-tests, failure of integrity check of the TSF executable image, failure of noise source health tests.

5.3.7.2 FPT_SKP_EXT.1 Extended: Protection of TSF Data (for reading of all symmetric keys)

FPT_SKP_EXT.1.1 The TSF shall prevent reading of all pre-shared keys, symmetric keys, and private keys.

5.3.7.3 FPT_APW_EXT.1 Extended: Protection of Administrator Passwords

FPT_APW_EXT.1.1 The TSF shall store passwords in non-plaintext form.

FPT_APW_EXT.1.2 The TSF shall prevent the reading of plaintext passwords.

5.3.7.4 FPT_STM.1 Reliable time stamps

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

5.3.7.5 FPT_TST_EXT.1: TSF Testing

FPT_TST_EXT.1.1 The TSF shall run a suite of self tests during initial start-up (on power on) to demonstrate the correct operation of the TSF.

FPT_TST_EXT.1.2 The TSF shall provide the capability to verify the integrity of stored TSF executable code when it is loaded for execution through the use of the TSF-provided cryptographic service specified in FCS_COP.1(2).

5.3.7.6 FPT_TUD_(EXT).1 Extended: Trusted Update

FPT_TUD_(EXT).1.1 The TSF shall provide security administrators the ability to query the current version of the TOE firmware/software.

FPT_TUD_(EXT).1.2 The TSF shall provide security administrators the ability to initiate updates to TOE firmware/software.

FPT_TUD_(EXT).1.3 The TSF shall provide a means to verify firmware/software updates to the TOE using a digital signature mechanism and [published hash] prior to installing those updates.

5.3.7.7 FTA_SSL_EXT.1 TSF-initiated Session Locking

FTA_SSL_EXT.1.1 The TSF shall, for local interactive sessions, [

- terminate the session]

after a Security Administrator-specified time period of inactivity.

5.3.7.8 FTA_SSL.3 TSF-initiated Termination

FTA_SSL.3.1(1) Refinement: The TSF shall terminate **a remote** interactive session after a [*Security Administrator-configurable time interval of session inactivity*].

5.3.7.9 FTA_SSL.3 TSF-initiated Termination – VPN client

FTA_SSL.3.1(2) Refinement: The TSF shall terminate **a remote VPN client** session after a [*Administrator-configurable time interval of session inactivity*].

5.3.7.10 FTA_SSL.4 User-initiated Termination

FTA_SSL.4.1 The TSF shall allow Administrator-initiated termination of the Administrator's own interactive session.

5.3.7.11 FTA_TAB.1 Default TOE Access Banners

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

5.3.8 Trusted Path/Channels (FTP)

5.3.8.1 FTP_ITC.1 Inter-TSF trusted channel

FTP_ITC.1.1 Refinement: The TSF shall **use IPsec, and [SSH]** to provide a **trusted** communication channel between itself and **all 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 and detection of modification of the channel data.**

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

FTP_ ITC.1.3 The TSF shall initiate communication via the trusted channel for [*communications with the following:*

- external audit servers using IPsec,
- remote AAA servers using SSH or IPsec,
- remote VPN gateways/peers using IPsec,
- another instance of the TOE using SSH or IPsec].

5.3.8.2 FTP_TRP.1 Trusted Path

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

FTP_TRP.1.2 Refinement: The TSF shall permit **remote administrators** to initiate communication via the trusted path.

FTP_TRP.1.3 The TSF shall require the use of the trusted path for initial administrator authentication and all remote administration actions.

5.4 TOE SFR Dependencies Rationale for SFRs Found in NDPP

The NDPPv1.1 and VPN EPv1.1 contain all the requirements claimed in this Security Target. As such the dependencies are not applicable since the PP itself has been approved.

5.5 Security Assurance Requirements

5.5.1 SAR Requirements

The TOE assurance requirements for this ST are taken directly from the Common Criteria Version 3.1, Revision 4. The assurance requirements are summarized in the table below.

Table 17: Assurance Measures

Assurance Class	Components	Components Description
DEVELOPMENT	ADV_FSP.1	Basic Functional Specification
GUIDANCE DOCUMENTS	AGD_OPE.1	Operational user guidance
	AGD_PRE.1	Preparative User guidance
LIFE CYCLE SUPPORT	ALC_CMC.1	Labeling of the TOE
	ALC_CMS.1	TOE CM coverage
TESTS	ATE_IND.1	Independent testing - conformance
VULNERABILITY ASSESSMENT	AVA_VAN.1	Vulnerability analysis

5.5.2 Security Assurance Requirements Rationale

The Security Assurance Requirements (SARs) in this Security Target represent the SARs identified in the NDPPv1.1 and VPNEPv1.1. As such, the NDPP SAR rationale is deemed acceptable since the PP itself has been validated.

5.5.3 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 18: Assurance Measures

Component	How requirement will be met
ADV_FSP.1	The functional specification describes the external interfaces of the TOE; such as the means for a user to invoke a service and the corresponding response of those services. The description includes the interface(s) that enforces a security functional requirement, the interface(s) that supports the enforcement of a security functional requirement, and the interface(s) that does not enforce any security functional requirements. The interfaces are described in terms of their purpose (general goal of the interface), method of use (how the interface is to be used), parameters (explicit inputs to and outputs from an interface that control the behavior of that interface), parameter descriptions (tells what the parameter is in some meaningful way), and error messages (identifies the condition that generated it, what the message is, and the meaning of any error codes). The development evidence also contains a tracing of the interfaces to the SFRs described in this ST.
AGD_OPE.1	The Administrative Guide provides the descriptions of the processes and procedures of how the administrative users of the TOE can securely administer the TOE using the interfaces that provide the features and functions detailed in the guidance.
AGD_PRE.1	The Installation Guide describes 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.1	The Configuration Management (CM) document(s) describes how the consumer (end-user) of

Component	How requirement will be met
ALC_CMS.1	the TOE can identify the evaluated TOE (Target of Evaluation). The CM document(s), identifies the configuration items, how those configuration items are uniquely identified, and the adequacy of the procedures that are used to control and track changes that are made to the TOE. This includes details on what changes are tracked, how potential changes are incorporated, and the degree to which automation is used to reduce the scope for error.
ATE_IND.1	Cisco provided the TOE for testing and, in coordination with the evaluation team, determined that the TOE was suitable for testing. All information provided met the requirements for content and presentation of evidence and testing was successfully completed based upon the requirements of the PP and extended package.
AVA_VAN.1	Cisco provided the TOE for testing and it was determined to be suitable for completion of the requirements. All information provided met the requirements for content and presentation of evidence. The evaluation team conducted a public search of potential vulnerabilities and ensured no issues resulted in a potential risk to the end user(s).

6 TOE SUMMARY SPECIFICATION

6.1 TOE Security Functional Requirement Measures

Table 18 identifies and describes how the Security Functional Requirements identified in section 5 of this ST are met by the TOE.

Table 19 How TOE SFRs Measures

TOE SFRs	How the SFR is Met
Security Functional Requirements Drawn from NDPP	
FAU_GEN.1	The TOE is able to generate audit records that are stored internally within the TOE whenever an audited event occurs. The types of events that cause audit records to be generated include: startup of the audit mechanism, cryptography related events; events related to the enforcement of identification and authentication related events; and administrative actions. Each of the messages contains sufficient 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. To ensure audit records are generated for the required auditable events, the TOE must be configured in its evaluated configuration as specified in the AGD documents. This is to ensure that auditing is enabled so that the audit records are being generated for the required auditable events. If the command 'no logging on' is entered the TOE is deemed no longer in the evaluated configuration. The audit trail consists of the individual audit records; one audit record for each event that occurred. The audit record can contain up to 80 characters and a percent sign (%), which follows the time-stamp information. As noted above, the information includes at least all of the required information. Example audit events are included below: Nov 19 13:55:59: %CRYPTO-6-SELF_TEST_RESULT: Self test info: (Self test activated by user: lab) Nov 19 13:55:59: %CRYPTO-6-SELF_TEST_RESULT: Self test info: (Software checksum ... passed) Nov 19 13:55:59: %CRYPTO-6-SELF_TEST_RESULT: Self test info: (DES encryption/decryption ... passed) Nov 19 13:55:59: %CRYPTO-6-SELF_TEST_RESULT: Self test info: (3DES encryption/decryption ... passed) Nov 19 13:55:59: %CRYPTO-6-SELF_TEST_RESULT: Self test info: (SHA hashing ... passed) Nov 19 13:55:59: %CRYPTO-6-SELF_TEST_RESULT: Self test info: (AES encryption/decryption ... passed) In the above log events a date and timestamp is displayed as well as an event description "CRYPTO-6-SELF_TEST_RESULT: Self test info: (Self test)". The subject identity where a command is directly run by a user is displayed "user: lab." The outcome of the command is displayed: "passed" The logging buffer size can be configured from a range of 4096 (default) to 2147483647 bytes. It is noted, not make the buffer size too large because the router could run out of memory for other tasks. Use the show memory privileged EXEC command to view the free processor memory on the router. However, this value is the maximum available, and the buffer size should not be set to this amount.

<table>
<tr><th>TOE SFRs</th><th>How the SFR is Met</th></tr>
<tr><td></td><td>
The administrator can also configure a ‘configuration logger’ to keep track of configuration changes made with the command-line interface (CLI). The administrator can configure the size of the configuration log from 1 to 1000 entries (the default is 100). Refer to the Guidance documentation for configuration syntax and information.

The log buffer is circular, so newer messages overwrite older messages after the buffer is full. Administrators are instructed to monitor the log buffer using the show logging privileged EXEC command to view the audit records. The first message displayed is the oldest message in the buffer. There are other associated commands to clear the buffer, to set the logging level, etc; all of which are described in the Guidance documents and IOS CLI.

The logs can be saved to flash memory so records are not lost in case of failures or restarts. Refer to the Guidance documentation for configuration syntax and information.

The administrator can set the level of the audit records to be displayed on the console or sent to the syslog server. For instance all emergency, alerts, critical, errors, and warning message can be sent to the console as an immediate indicator of a generated syslog event. All notifications and information type message can be sent to the syslog server, as message is only for information; router functionality is not affected.

The only time dropped packets would not be audited is when the router is booting or if the logging of dropped packets has not been enabled or has been specifically disabled. During the boot process, all packets are automatically dropped.

<table>
<tr><th>Auditable Event</th><th>Rationale</th></tr>
<tr><td>All use of the user identification mechanism.</td><td>Events will be generated for attempted identification/ authentication, and the username attempting to authenticate and source address or interface will be included in the log record.</td></tr>
<tr><td>Any use of the authentication mechanism.</td><td>Events will be generated for attempted identification/ authentication, and the username attempting to authenticate will be included in the log record, along with the origin or source of the attempt.</td></tr>
<tr><td>Management functions</td><td>The use of the security management functions is logged; modifications of the behavior of the functions in the TSF and modifications of default settings.</td></tr>
<tr><td>Changes to the time.</td><td>Changes to the time are logged.</td></tr>
<tr><td>Failure to establish and/or establishment/failure of an IPSEC session</td><td>Attempts to establish an IPSEC session or the failure of an established IPSEC tunnel is logged as well as successfully established and terminated IPsec sessions.</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>TOE SFRs</th><th>How the SFR is Met</th></tr>
<tr><td></td><td>
<table>
<tr><td>Establishing session with CA</td><td>The connection to CA’s for the purpose of certificate verification is logged.</td></tr>
<tr><td>Failure to establish and/or establishment/failure of an SSH session</td><td>Attempts to establish an SSH session or the failure of an established SSH is logged as well as successfully established and terminated sessions.</td></tr>
<tr><td>Application of rules configured with the ‘log’ operation</td><td>Logs are generated when traffic matches acls that are configured with the log operation.</td></tr>
<tr><td>Indication of packets dropped due to too much network traffic</td><td>Logs are generated when traffic that exceeds the settings allowed on an interface is received.</td></tr>
<tr><td>Indication that TSF self-test was completed.</td><td>During bootup, if the self test fails, the failure is logged.</td></tr>
<tr><td>Initiation of update</td><td>Audit event is generated for the initiation of a software update.</td></tr>
<tr><td>Any attempts at unlocking of an interactive session.</td><td>Audit event is generated after a user’s session is locked and the admin user is required to re-authenticate.</td></tr>
<tr><td>Once a remote interactive session is terminated after a Security Administrator-configurable time interval of session inactivity.</td><td>An audit event is generated by the termination of a remote session by the session locking mechanism.</td></tr>
<tr><td>The termination of an interactive session.</td><td>An audit event is generated by an authorized administrator when the exit command is used.</td></tr>
<tr><td>Initiation of the trusted channel/ path.
Termination of the trusted channel/ path.
Failure of the trusted channel/ path functions.</td><td>See the rows for IPsec and SSH above.</td></tr>
</table>
</td></tr>
<tr><td>FAU_GEN.2</td><td>The TOE shall ensure that each auditable event is associated with the user that triggered the event and as a result they are traceable to a specific user. For example a human user, user identity, or related session ID would be included in the audit record. For an IT entity or device, the IP address, MAC address, host name, or other configured identification is presented. A sample audit record is below:

Jun 18 11:17:20.505: SSH0: protocol version id is - SSH-2.0-Cisco-1.25
Jun 18 11:17:20.769: AAA/BIND(0000004B): Bind i/f
Jun 18 11:17:20.769: AAA/AUTHEN/LOGIN (0000004B): Pick method list 'default'
Jun 18 2012 11:17:26 UTC: %SEC_LOGIN-5-LOGIN_SUCCESS: Login Success</td></tr>
</table>

TOE SFRs	How the SFR is Met
	[user: admin] [Source: 100.1.1.5] [localport: 22] at 11:17:26 UTC Mon Jun 18 2012
FAU_STG_EXT.1	The TOE is configured to export syslog records to a specified, external syslog server. The TOE protects communications with an external syslog server via IPsec. The TOE transmits its audit events to all configured syslog servers at the same time logs are written to the local log buffer and to the console. The TOE is capable of detecting when the IPsec connection fails. The TOE also stores a limited set of audit records locally on the TOE, and continues to do so if the communication with the syslog server goes down. If the IPsec connection fails, the TOE will buffer a small amount of audit records on the TOE when it discovers it can no longer communicate with its configured syslog server, and will transmit the buffer contents when connectivity to the syslog server is restored. This store is separate from the local logging buffer, which could be set to a different level of logging then what is to be sent via syslog. Only Authorized Administrators are able to clear the local logs, and local audit records are stored in a directory that does not allow administrators to modify the contents. In addition to the remote syslog view of the logs, logs may be viewed locally on the TOE by issuing the “show logging” command either locally or remotely (via SSH).
FCS_CKM.1(1) FCS_CKM.1(2)	The TOE implements a random number generator for Diffie-Hellman and Elliptic curve based key establishment (conformant to NIST SP 800-56A), and for RSA key establishment schemes (conformant to NIST SP 800-56B). The TOE can create a RSA public-private key pair that can be used to generate a Certificate Signing Request (CSR). Through use of Simple Certificate Enrollment Protocol (SCEP), the TOE can: send the CSR to a Certificate Authority (CA) for the CA to generate a certificate; and receive its certificate (including X.509v3) from the CA. Integrity of the CSR and certificate during transit are assured through use of digitally signatures (encrypting the hash of the TOE’s public key contained in the CSR and certificate). The TOE can store and distribute the certificate to external entities including Registration Authorities (RA). The IOS Software supports embedded PKI client functions that provide secure mechanisms for distributing, managing, and revoking certificates. In addition, the IOS Software includes an embedded certificate server, allowing the router to act as a certification authority on the network. The TOE can also use the X.509v3 certificate for securing IPsec and SSH, sessions.
FCS_CKM_EXT.4	The TOE meets all requirements specified in FIPS 140-2 for destruction of keys and Critical Security Parameters (CSPs) in that none of the symmetric keys, pre-shared keys, or private keys are stored in plaintext form. See Table 19 for more information on the key zeroization.
FCS_COP.1(1)	The TOE provides symmetric encryption and decryption capabilities using AES in CBC and GCM mode (128, 256 bits) as described in NIST SP 800-38A and NIST SP 800-38D.
FCS_COP.1(2)	The TOE will provide cryptographic signature services using RSA with key size of 2048 and greater as specified in FIPS PUB 186-3, “Digital Signature Standard” and FIPS PUB 186-2, “Digital Signature Standard”. In addition, the TOE will provide cryptographic signature services using ECDSA with key size of 256 and greater as specified in FIPS PUB 186-3, “Digital Signature Standard”.
FCS_COP.1(3)	The TOE provides cryptographic hashing services using SHA-1, SHA-256, SHA-384, and SHA-512 as specified in FIPS Pub 180-3 “Secure Hash Standard.”
FCS_COP.1(4)	The TOE provides keyed-hashing message authentication services using HMAC-SHA-1 as specified in FIPS Pub 198-1,"The Keyed-Hash Message Authentication Code,” and FIPS 180-3, “Secure Hash Standard.”
FCS_IPSEC_EXT.1	The IPsec implementation provides both VPN peer-to-peer and VPN client to TOE

TOE SFRs	How the SFR is Met
	capabilities. The VPN peer-to-peer tunnel allows for example the TOE and another router to establish an IPsec tunnel to secure the passing of route tables (user data). Another configuration in the peer-to-peer configuration is to have the TOE be set up with an IPsec tunnel with a VPN peer to secure the session between the TOE and syslog server. The VPN client to TOE configuration would be where a remote VPN client connects into the TOE in order to gain access to an authorized private network. Authenticating with the TOE would give the VPN client a secure IPsec tunnel to connect over the internet into their private network. In addition to tunnel mode, which is the default IPSec mode, the TOE also supports transport mode, allowing for only the payload of the packet to be encrypted. If tunnel mode is explicitly specified, the router will request tunnel mode and will accept only tunnel mode. The TOE implements IPsec to provide both certificates and pre-shared key-based authentication and encryption services to prevent unauthorized viewing or modification of data as it travels over the external network. The TOE implementation of the IPsec standard (in accordance with the RFCs noted in the SFR) uses the Encapsulating Security Payload (ESP) protocol to provide authentication, encryption and anti-replay services. Preshared keys can be configured using the 'crypto isakmp key' key command and may be proposed by each of the peers negotiating the IKE establishment. IPsec Internet Key Exchange, also called ISAKMP, is the negotiation protocol that lets two peers agree on how to build an IPsec Security Association (SA). The IKE protocols implement Peer Authentication using the RSA, ECDSA algorithm with X.509v3 certificates, or preshared keys. IKE separates negotiation into two phases: phase 1 and phase 2. Phase 1 creates the first tunnel, which protects later ISAKMP negotiation messages. The key negotiated in phase 1 enables IKE peers to communicate securely in phase 2. During Phase 2 IKE establishes the IPsec SA. IKE maintains a trusted channel, referred to as a Security Association (SA), between IPsec peers that is also used to manage IPsec connections, including: • The negotiation of mutually acceptable IPsec options between peers (including peer authentication parameters, either signature based or pre-shared key based), • The establishment of additional Security Associations to protect packets flows using Encapsulating Security Payload (ESP), and • The agreement of secure bulk data encryption AES keys for use with ESP. After the two peers agree upon a policy, the security parameters of the policy are identified by an SA established at each peer, and these IKE SAs apply to all subsequent IKE traffic during the negotiation. The TOE supports both IKEv1 and IKEv2 session establishment. As part of this support, the TOE can be configured to not support aggressive mode for IKEv1 exchanges and to only use main mode using the 'crypto isakmp aggressive-mode disable' command. The TOE can be configured to not allow "confidentiality only" ESP mode by ensuring the IKE Policies configured include ESP-encryption. The TOE supports configuration lifetimes of both Phase 1 SAs and Phase 2 SAs using "lifetime" command. The default time value for Phase 1 SAs is 24 hours. The default time value for Phase 2 SAs is 1 hour, but it is configurable to 8 hours. The TOE supports configuring the maximum amount of traffic that is allowed to flow for a given IPsec SA using the following command, 'crypto ipsec security-association lifetime'. The default amount is 2560KB, which is the minimum configurable value. The maximum configurable value is 4GB.

TOE SFRs	How the SFR is Met
	The TOE provides AES-GCM-128, AES_GCM-256, AES-CBC-128, and AES-CBC-256 for encrypting the IKEv1 and IKEv2 payloads. The administrator is instructed in the AGD to ensure that the size of key used for ESP must be greater than or equal to the key size used to protect the IKE payload. The TOE supports Diffie-Hellman Group 14 (2048-bit keys), 19 (256-bit Random ECP), 24 (2048-bit MODP with 256-bit POS), 20 (384-bit Random ECP), 15 (3072 bit MODP), and 16 (4096-bit MODP) in support of IKE Key Establishment. These keys are generated using the AES-CTR Deterministic Random Bit Generator (DRBG), as specified in SP 800-90, and the following corresponding key sizes (in bits) are used: 320 (for DH Group 14), 256 (for DH Group 19), 256 (for DH Group 24), 384 (for DH Group 20), 424 (for DH Group 15), and 480 (bits for DH Group 16)] bits IPsec provides secure tunnels between two peers, such as two routers and remote VPN clients. An authorized administrator defines which packets are considered sensitive and should be sent through these secure tunnels. When the IPsec peer recognizes a sensitive packet, the peer sets up the appropriate secure tunnel and sends the packet through the tunnel to the remote peer. More accurately, these tunnels are sets of security associations (SAs) that are established between two IPsec peers or between the TOE and remote VPN client. The SAs define the protocols and algorithms to be applied to sensitive packets and specify the keying material to be used. SAs are unidirectional and are established per security protocol (AH or ESP). In the evaluated configuration only ESP will be configured for use. A crypto map (the Security Policy Definition) set can contain multiple entries, each with a different access list. The crypto map entries are searched in a sequence - the router attempts to match the packet to the access list (acl) specified in that entry. When a packet matches a permit entry in a particular access list, the method of security in the corresponding crypto map is applied. If the crypto map entry is tagged as ipsecisakmp, IPsec is triggered. The traffic matching the permit acls would then flow through the IPSec tunnel and be classified as "PROTECTED". Traffic that does not match a permit acl in the crypto map, but that is not disallowed by other acls on the interface is allowed to BYPASS the tunnel. Traffic that does not match a permit acl and is also blocked by other non-crypto acls on the interface would be DISCARDED. The TOE implementation of the IPsec standard (in accordance with the RFCs noted in the SFR and using cryptographic algorithms AES-GCM-128, AES-GCM-256, AES-CBC-128 and AES-CBC-256 together with HMAC-SHA1) uses the Encapsulating Security Payload (ESP) protocol to provide authentication, encryption and anti-replay services. If there is no SA that the IPsec can use to protect this traffic to the peer, IPsec uses IKE to negotiate with the remote peer to set up the necessary IPsec SAs on behalf of the data flow. The negotiation uses information specified in the crypto map entry as well as the data flow information from the specific access list entry.
FCS_SSH_EXT.1	The TOE implements SSHv2 (telnet is disabled in the evaluated configuration). SSHv2 session establishment is limited to a configurable session timeout period of up to 120 seconds, and a maximum number of failed authentication attempts limited to 3. SSH connections will be dropped if the TOE receives a packet larger than 35,000 bytes. Large packets are detected by the SSH implementation, and dropped internal to the SSH process. The DH group 14 is a configurable option in the TOE. The TOE implementation of SSHv2 supports the following: • public key algorithms for authentication: RSA Signature Verification. • local password-based authentication for administrative users accessing the TOE through SSHv2, and optionally supports deferring authentication to a remote AAA server.

TOE SFRs	How the SFR is Met
	• encryption algorithms, AES-CBC-128, AES-CBC-256 to ensure confidentiality of the session. • hashing algorithms HMAC-SHA1, HMAC-SHA1-96 to ensure the integrity of the session. • Diffie-Hellman Group 14 (2048-bit keys) Key Establishment, as required by the PP.
FCS_RBG_EXT.1	The TOE implements a NIST-approved AES-CTR Deterministic Random Bit Generator (DRBG), as specified in SP 800-90. The DRBG is supplied with entropy from a hardware based source. The hardware based entropy source used is an on-board chip that uses analog ring oscillator based noise to produce random output that is made available through a register. Ring oscillators produce fluctuating output. Due to the effect of several forms of electronic noise, primarily thermal noise, the ring oscillator output signal transitions before or after the expected switching time. This effect is referred to as 'Ring Oscillator Jitter' in the time domain and as phase noise in the frequency domain. The ring oscillator based entropy source used on the platforms being tested was found to generate output that possesses a substantially high amount of entropy The ring oscillator operates independently, and the entropy source is protected within the boundary of the TOE. An adversary on the outside is not able to affect the entropy rate in any determinable way, because of the number of sources, and the fact that the only one of the sources (allocated packet buffer) is populated with data that came from outside of the system.
FDP_RIP.2	The TOE ensures that packets transmitted from the TOE do not contain residual information from previous packets. Frames that are not the required length use zeros for padding. Residual data is never transmitted from the TOE. Once packet handling is completed memory buffer content is zeroized before reuse. This applies to both data plane traffic and administrative session traffic.
FIA_AFL.1	The TOE provides the privileged administrator the ability to specify the maximum number of unsuccessful authentication attempts (between 1 and 25) before privileged administrator or non-privileged administrator is locked out through the administrative CLI using a privileged CLI command. When a privileged administrator or non-privileged administrator attempting to log into the administrative CLI reaches the administratively set maximum number of failed authentication attempts, the user will not be granted access to the administrative functionality of the TOE until a privileged administrator resets the user's number of failed login attempts through the administrative CLI.
FIA_PMG_EXT.1	The TOE supports the local definition of users with corresponding passwords. The passwords can be composed of any combination of upper and lower case letters, numbers, and special characters (that include: "!", "@", "#", "$", "%", "^", "&", "*", "(", and ")". Minimum password length is settable by the Authorized Administrator, and can be configured for minimum password lengths of up 15 characters.
FIA_PSK_EXT.1	The TOE supports use of IKEv1 (ISAKMP) and IKEv2 pre-shared keys for authentication of IPsec tunnels. Preshared keys can be entered as ASCII character strings, or HEX values. The TOE supports keys that are from 22 characters in length up to 128 bytes in length. The data that is input is conditioned prior to use via SHA-1 or AES.
FIA_UIA_EXT.1	The TOE requires all users to be successfully identified and authenticated before allowing any TSF mediated actions to be performed. Administrative access to the TOE is facilitated through the TOE's CLI. The TOE mediates all administrative actions through the CLI. Once a potential administrative user attempts to access the

TOE SFRs	How the SFR is Met
	CLI 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 functionality of the TOE until an administrator is successfully identified and authenticated.
FIA_UAU_EXT.2	The TOE provides a local password-based authentication mechanism as well as support for RADIUS and TACACS+ authentication. The administrator authentication policies include authentication to the local user database or redirection to a remote authentication server. Interfaces can be configured to try one or more remote authentication servers, and optionally fallback to the local user database if the remote authentication servers are inaccessible. The TOE correctly invokes an external authentication server to provide a remote authentication mechanism, or password-based authentication by forwarding the authentication requests to the external authentication server. The process for authentication is the same for administrative access whether administration is occurring via a directly connected console cable or remotely via SSHv2. At initial login in the administrative user is prompted to provide a username. After the user provides the username, the user is prompted to provide the administrative password associated with the user account. The TOE then either grants administrative access (if the combination of username and password is correct) or indicates that the login was unsuccessful. The TOE does not provide a reason for failure in the cases of a login failure.
FIA_UAU.7	When a user enters their password at the local console or via SSH, the TOE does not echo any characters of the password or any representation of the characters.
FIA_X509_EXT.1	The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for IPsec and SSH connections. Public key infrastructure (PKI) credentials, such as Rivest, Shamir, and Adelman (RSA) keys and certificates can be stored in a specific location on the router, such as NVRAM and flash memory or on a USB eToken 64 KB smart card. The certificates are manually loaded on the TOE via an authorized administrator using the CLI. The certificates themselves provide protection in that they are digitally signed. If a certificate is modified in any way, it would be invalidated. The digital signature verifications process would show that the certificate had been tampered with when the hash value would be invalid. The physical security of the router (A.Physical) protects the router and the certificates from being tampered with or deleted. In addition, the TOE identification and authentication security functions protect an unauthorized user from gaining access to the TOE. USB tokens provide for secure configuration distribution of the digital certificates and private keys. RSA operations such as on-token key generation, signing, and authentication, and the storage of Virtual Private Network (VPN) credentials for deployment can be implemented using the USB tokens. Both OSP and CRL are configurable and may be used for certificate revocation. Checking is also done for the basicConstraints extension and the cA flag to determine whether they are present and set to TRUE. If they are not, the certificate is not accepted.
FMT_MOF.1	The TOE provides the ability for Authorized Administrators to access TOE data, such as audit data, configuration data, security attributes, information flow rules, routing tables, and session thresholds. Each of the predefined and administratively configured privilege level has a set of permissions that will grant them access to the TOE data, though with some privilege levels, the access is limited. The TOE performs role-based authorization, using TOE platform authorization mechanisms, to grant access to the privilege levels. For the purposes of this evaluation, the Authorized Administrator is an authenticated administrator whose current privilege level is sufficient to perform the desired administrative actions. Privilege levels 0

TOE SFRs	How the SFR is Met
FMT_MTD.1	and 1 are defined by default and are customizable, while levels 2-14 are undefined by default and are also customizable. Privilege level 15 is not customizable, and provides full access to all administrative functions. When a username has a privilege level assigned to it, the level defines the highest privilege level accessible with that username's credentials. All new administrative sessions start at privilege level 1 regardless of the privilege level assigned to the username. Authenticated administrators can use the "enable" command to switch from privilege level 1 to their highest allowed privilege level using their own password, or can use "enable ##" to switch to another privilege level if an "enable password" has been configured for the level. The term "Authorized Administrator" is used in this ST to refer to any user who has been assigned to a privilege level that is permitted to perform the relevant action; therefore, the user has the appropriate privileges to perform the requested functions. Semi-privileged administrators with only a subset of privileges can also modify TOE data based on if granted the privilege.
FMT_SMF.1	The TOE provides all the capabilities necessary to securely manage the TOE. The administrative user can connect to the TOE using the CLI to perform these functions via SSHv2, a terminal server, or at the local console. The specific management capabilities available from the TOE include: • Local and remote administration of the TOE and the services provided by the TOE via the TOE CLI, as described above; • The ability to update the IOS software (image integrity verification is provided using SHA-256) • Ability to configure the cryptographic functionality; • Ability to configure the IPsec functionality, • Ability to enable, disable, determine and modify the behavior of all the security functions of the TOE via the CLI.
FMT_SMR.2	The TOE platform maintains privileged and semi-privileged administrator roles. The TOE performs role-based authorization, using TOE platform authorization mechanisms, to grant access to the semi-privileged and privileged roles. For the purposes of this evaluation, the privileged role is equivalent to full administrative access to the CLI, which is the default access for IOS privilege level 15; and the semi-privileged role equates to any privilege level that has a subset of the privileges assigned to level 15. Privilege levels 0 and 1 are defined by default and are customizable, while levels 2-14 are undefined by default and are also customizable. Note: the levels are not hierarchical. The term "Authorized Administrator" is used in this ST to refer to any user which has been assigned to a privilege level that is permitted to perform the relevant action; therefore has the appropriate privileges to perform the requested functions. The privilege level determines the functions the user can perform; hence the Authorized Administrator with the appropriate privileges. The TOE can and shall be configured to authenticate all access to the command line interface using a username and password. The TOE supports both local administration via a directly connected console cable and remote authentication via SSH.
FPT_FLS.1	Whenever a 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

TOE SFRs	How the SFR is Met
	information to or from the TOE and reloads. So long as the failures persist, the TOE will continue to reload. This functionally prevents any failure from causing an unauthorized information flow. There are no failures that circumvent this protection.
FPF_RUL_EXT.1	An authorized administrator can define the traffic that needs to be tunneled by configuring access lists (permit, deny, log) and applying these access lists to interfaces using access-lists and crypto map. The rules (access control list entries) within an access-list applied to a crypto map determine whether the traffic is to be encrypted/decrypted. If traffic is not defined within the crypto map access-list, encryption/decryption will not be applicable to that traffic flow, and access-lists applied to interfaces (not to crypto maps) will determine whether the traffic will be permitted/denied through the TOE (without encryption/decryption by the TOE). Traffic may be selected on the basis of the source and/or destination address, and optionally the IP protocol and/or Layer 4 protocol and port. The TOE enforces information flow policies on network packets that are received by TOE interfaces and leave the TOE through other TOE interfaces. The TOE interfaces are the external network Ethernet ports that network traffic within the suite of the IP protocol family traverses. All such data is subject to internal filtering rules which restrict the flow of Layer 3 network traffic to and from each TOE interface. The TOE does not support VLAN interfaces, so all tunnel endpoints on the TOE will be individual physical interfaces, not a set of multiple physical interfaces through which an VLAN interface is accessed. When network packets are received on a TOE interface, the TOE verifies whether the network traffic is allowed or not and performs one of the following actions, pass/not pass information, as well as optional logging. These rules control whether a packet is transferred from one interface to another based on: 1. IP address of source (as defined in the packet header) 2. IP address of destination (as defined in the packet header) 3. IP protocol 4. transport layer protocol (or next header in IPv6) 5. Service used (UDP or TCP ports, both source and destination) 6. Network interface on which the connection request occurs These rules are supported for the following protocols: RFC 791(IPv4); RFC 2460 (IPv6); RFC 793 (TCP); RFC 768 (UDP). TOE compliance with these protocols is verified via regular quality assurance, regression, and interoperability testing. These rules are operational as soon as interfaces are operational following the startup of the TOE. There is no state during initialization/startup that the access-lists are not enforced on an interface. By default, packets will not be encrypted/decrypted unless a specific rule matching that traffic flow has been applied to a crypto map, and that crypto map applied to the applicable interface. Whether not packets are match a crypto map, the packets need to be permitted by any access-list applied to inbound or outbound interfaces. Rules are enforced on a first match basis from the top down. As soon as a match is found the action associated with the rule is applied. These rules are entered in the form of access lists at the CLI via the ‘access-list’ command, and applied to interfaces via the ‘access group’ command or the ‘match’ command. These rules are operational as soon as interfaces are operational following startup of the TOE. There is no state during initialization/ startup that the access lists are not enforced on an interface. In the event of failure to the router (including memory buffer overflow or failure of a

TOE SFRs	How the SFR is Met
	self-test), network traffic would not be forwarded. In the event of failure of one or more network interfaces, traffic would not flow across the failed interface, but would continue to flow across other interfaces as long as no failed interfaces is part of the traffic path through the TOE. Otherwise, these interfaces pass traffic only when its source address matches the network interface originating the traffic through another network interface corresponding to the traffic's destination address. Indication of the traffic dropped due to too much network traffic is provided by an audit event. During the boot cycle, the TOE first powers on the hardware, loads the image, and executes the power on self-tests. Until the power on self-tests successfully complete, the interfaces to the TOE are deactivated. Once the tests complete, the interfaces become active and the rules associated with the interface become immediately operational. By default, no traffic passes until access-lists that allow traffic to pass are assigned to the TOE's interfaces and the TOE is operational in Normal mode of operation. There is no state during initialization/ startup that the access lists are not enforced on an interface.
FPT_SKP_EXT.1	The TOE stores all private keys in a secure directory that is not readily accessible to administrators. All pre-shared and symmetric keys are stored in encrypted form using AES encryption to additionally obscure access. This functionality is configured on the TOE using the 'password encryption aes' command. The TOE is configured to not display configured keys as part of configuration files using the 'hidekeys' command.
FPT_APW_EXT.1	The TOE includes a Master Passphrase feature that can be used to configure the TOE to encrypt all locally defined user passwords. In this manner, the TOE ensures that plaintext user passwords will not be disclosed even to administrators.
FPT_STM.1	The TOE provides a source of date and time information used in audit event timestamps. The clock function is reliant on the system clock provided by the underlying hardware. The TOE can optionally be set to receive clock updates from an NTP server. This date and time is used as the time stamp that is applied to TOE generated audit records and used to track inactivity of administrative sessions.
FPT_TUD_EXT.1	The TOE has specific versions that can be queried by an administrator. When updates are made available by Cisco, an administrator can obtain and install those updates. The updates can be downloaded from the Cisco.com web site. Authorized Administrators can download the Common Criteria evaluated software image file from Cisco.com onto a trusted computer system for usage in the trusted update functionality. Software images are available from Cisco.com at the following: http://www.cisco.com/cisco/software/navigator.html. The cryptographic checksums (i.e., public hashes/SHA-256) are used to verify software/firmware update files (to ensure they have not been modified from the originals distributed by Cisco) before they are used to actually update the applicable TOE components. Once the file is downloaded from the Cisco.com web site, verify that it was not tampered. The verification is done by using a hash utility to compute a hash value for the downloaded file and comparing this with the hash value for the image. The hash value can be displayed by hovering over the software image name under details on the Cisco.com web site. If the hashes do not match, contact Cisco Technical Assistance Center (TAC). If a digital signature is used for verification, the certificate issued to the TOE needs to be issued from a trusted external trusted Certification Authority such as ex. Verisign or Entrust or must be from a trusted internal Certification Authority from within the TOE administrator's company or a self-signed certificate generated on the TOE itself. Certificates are stored to NVRAM by default; however, some routers do not have the required amount of NVRAM to successfully store certificates. All Cisco

TOE SFRs	How the SFR is Met
	platforms support NVRAM and flash local storage. Depending on the platform, an authorized administrator may have other supported local storage options including bootflash, slot, disk, USB flash, or USB token. During run time, an authorized administrator can specify what active local storage device will be used to store certificates. When a valid image is installed on the TOE, the digital signature will be validated and the image will be successfully installed. When an invalid image is attempted to be installed, the TOE will display an error and will reject the image as an invalid or corrupt image.
FPT_TST_EXT.1	As a FIPS 140-2 validated product, the TOE runs a suite of self-tests during initial start-up to verify its correct operation. If any of the tests fail, the Authorized Administrator will have to log into the CLI to determine which test failed and why. If the tests pass successfully the POST event logs will show successful for each test. During the system bootup process (power on or reboot), all the Power on Startup Test (POST) components for all the cryptographic modules perform the POST for the corresponding component (hardware or software). These tests include: • AES Known Answer Test • RSA Signature Known Answer Test (both signature/verification) • Power up bypass test • RNG Known Answer Test • Diffie Hellman test • HMAC Known Answer Test • SHA-1/256/512 Known Answer Test • Triple-DES Known Answer Test • Software Integrity Test If any component reports failure for the POST, the system crashes and appropriate information is displayed on the screen, and saved in the crashinfo file. All ports are blocked from moving to forwarding state during the POST. If all components of all modules pass the POST, the system is placed in FIPS PASS state and ports are allowed to forward data traffic. These tests are sufficient to verify that the correct version of the TOE software is running as well as that the cryptographic operations are all performing as expected.
FTA_SSL_EXT.1 FTA_SSL.3(1)	An administrator can configure maximum inactivity times individually for both local and remote administrative sessions through the use of the "session-timeout" setting applied to the console and virtual terminal (vty) lines. The configuration of the vty lines sets the configuration for the remote console access. The line console settings are not immediately activated for the current session. When a session is inactive (i.e., no session input from the administrator) for the configured period of time the TOE will terminate the session, and no further activity is allowed requiring the administrator to log in (be successfully identified and authenticated) again to establish a new session. Administratively configurable timeouts are also available for the EXEC level access (access above level 1) through use of the "exec-timeout" setting.
FTA_SSL.3(2)	When a remote VPN client session reaches a period of inactivity, its connection is terminated and it must re-establish the connection with new authentication to resume operation. This period of inactivity is set by the administrator in the VPN configuration.
FTA_SSL.4	An administrator is able to exit out of both local and remote administrative sessions.
FTA_TAB.1	The TOE displays a privileged Administrator specified banner on the CLI management interface prior to allowing any administrative access to the TOE. This

TOE SFRs	How the SFR is Met
	is applicable for both local and remote TOE administration.
FTA_TSE.1	The TOE allows for creation of acls that restrict vpn connectivity based on time, day, and the client's IP address (location). These acls allow customization of all of these properties to allow or deny access.
FTA_VCM_EXT.1	The TOE provides the option to use Network Address Translation to assign the remotely connecting VPN client an internal network IP address.
FTP_ITC.1	The TOE protects communications with peer or neighbor routers using keyed hash as defined in FCS_COP.1.1(4) and cryptographic hashing functions FCS_COP.1.1(3). This protects the data from modification of data by hashing that verify that data has not been modified in transit. In addition, encryption of the data as defined in FCS_COP.1.1(1) is provided to ensure the data is not disclosed in transit. The TOE also requires that peers and other TOE instances establish an IKE/IPsec connection in order to forward routing tables used by the TOE. In addition the TOE can establish secure VPN tunnels with IPsec VPN clients. The TOE can also secure communication with other instances of the TOE using SSH. The distinction between "remote VPN gateway/peer" and "another instance of the TOE" is that "another instance of the TOE" would be installed in the evaluated configuration, and likely administered by the same personnel, whereas a "remote VPN gateway/peer" could be any interoperable IPsec gateway/peer that is expected to be administered by personnel who are not administrators of the TOE, and who share necessary IPsec tunnel configuration and authentication credentials with the TOE administrators. For example, the exchange of X.509 certificates for certificate based authentication.
FTP_TRP.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 TOE can also be configured to use IPsec to secure administrative communications.

6.2 Key Zeroization

The following table describes the key zeroization referenced by FCS_CKM_EXT.4 provided by the TOE.

Table 20: TOE Key Zeroization

Name	Description	Zeroization
Diffie-Hellman Shared Secret	Shared secret generated by the Diffie-Hellman Key exchange	Automatically after session is terminated Overwrtten with: 0x00
Diffie Hellman private exponent	The private exponent used in Diffie-Hellman (DH) exchange. Generate by the module. Zeroized after DH shared secret has been generated.	Automatically after shared secret generated. Overwrtten with: 0x00
skeyid	Value derived from the shared secret within IKE exchange. Zeroized when IKE session is terminated.	Automatically after IKE session terminated. Overwrtten with: 0x00

Name	Description	Zeroization
skeyid_d	The IKE key derivation key for non ISAKMP security associations.	Automatically after IKE session terminated. Overwrtten with: 0x00
IKE session encrypt key	The IKE session encrypt key. Generate by the module	Automatically after IKE session terminated. Overwrtten with: 0x00
IKE session authentication key	The IKE session authentication key. Generate by the module.	Automatically after IKE session terminated. Overwrtten with: 0x00
ISAKMP preshared	The key used to generate IKE skeyid during preshared-key authentication. It is entered by the Crypto Officer. “no crypto isakmp key” command zeroizes it. This key can have two forms based on whether the key is related to the hostname or the IP address.	Zeroized using the following command: # no crypto isakmp key Overwrtten with: 0x0d
IKE RSA Private Key	RSA private key for IKE authentication. Generated or entered like any RSA key, set as IKE RSA Authentication Key with the “crypto keyring” or “ca trust-point” command.	Zeroized using the following command: # crypto key zeroize rsa Overwrtten with: 0x0d
IPsec encryption key	The IPSec encryption key. Generate by the module. Zeroized when IPSec session is terminated.	Automatically when IPsec session terminated. Overwrtten with: 0x00
IPsec authentication key	The IPSec authentication key. Generate by the module. The zeroization is the same as above.	Automatically when IPsec session terminated. Overwrtten with: 0x00
RADIUS secret	The RADIUS shared secret. It is entered by the Crypto Officer. This shared secret is zeroized by executing the “no radius-server key” command.	Zeroized using the following command: # no radius-server key Overwrtten with: 0x0d
TACACS+ secret	The TACACS+ shared secret. It is entered by the Crypto Officer. This shared secret is zeroized by executing the “no tacacs-server key” command.	Zeroized using the following command: # no tacacs-server key Overwrtten with: 0x0d
SSH Private Key	This key is used for message signing when performing SSH authentication. Generated by the module.	Zeroized using the following command: # crypto key zeroize rsa Overwrtten with: 0x00

Name	Description	Zeroization
SSH Session Key	This is the SSH session key. It is used to encrypt all SSH data traffics traversing between the SSH client and SSH server. It is generated by the module	Automatically when the SSH session is terminated. Overwrtten with: 0x0d

7 ANNEX A: REFERENCES

The following documentation was used to prepare this ST:

Table 21: References

Identifier	Description
[CC_PART1]	Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and general model, dated September 2012, version 3.1, Revision 4, CCMB-2012-009-001
[CC_PART2]	Common Criteria for Information Technology Security Evaluation – Part 2: Security functional components, dated September 2012, version 3.1, Revision 4, CCMB-2012-009-002
[CC_PART3]	Common Criteria for Information Technology Security Evaluation – Part 3: Security assurance components, dated September 2012, version 3.1, Revision 4, CCMB-2012-009-003
[CEM]	Common Methodology for Information Technology Security Evaluation – Evaluation Methodology, dated September 2012, version 3.1, Revision 4, CCMB-2012-009-004