Cisco 5921 Embedded Services Router

Security Target

Version 1.0

June 22, 2015

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
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
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 5921 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	5921 Embedded Services Router
ST Version	1.0
Publication Date	June 22, 2015
Vendor and ST Author	Cisco Systems, Inc.
TOE Reference	5921 Embedded Services Router
TOE Software Version	IOS 15.5(2)T
Keywords	Data Protection, Authentication, vpn

1.2 TOE Overview

The Cisco 5921 Embedded Services Router (ESR) running IOS 15.5(2)T (herein after referred to as the ESR, the vpn client, or the TOE). The TOE is a software-only Linux based router application. For the purposes of this evaluation, the TOE is a vpn client application.

1.2.1 TOE Product Type

The Cisco 5921 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 session capabilities.

The ESR is a software-only solution for protecting the network. The focus of this evaluation is on the IPsec Virtual Private Network (VPN) client that is part of the Cisco 5921 Embedded Services Router. The ESR provides vpn client session capabilities that provide secure tunnels to authenticated remote endpoints or gateways. The ESR encrypts all information that flows between itself and its VPN gateway or IPsec peer.

1.2.2 Supported non-TOE / Software/ Firmware

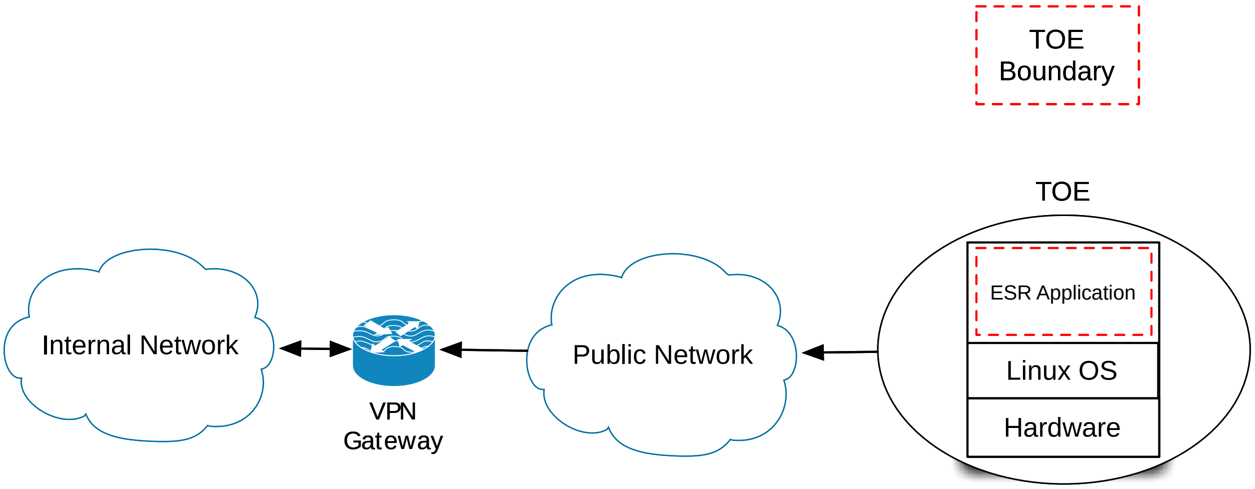
The TOE supports (in some cases optionally) the following hardware, software, and firmware in its environment when the TOE is configured in its evaluated configuration:

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.
Hardware	Yes	The TOE relies on underlying Hardware. A minimum of 256 MB memory is required and x86 processors.
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	No	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.
Operating System	Yes	The TOE relies on an underlying Linux Operating System. Linux Kernel 2.6.32 or later version is required.
IPsec Peer	Yes	This includes any IPsec peer with which the TOE participates in a secure IPsec session.
VPN Gateway	Yes	This includes any IPsec VPN Gateway with which the TOE participates in a secure IPsec session.
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 5921 Embedded Services Router Target of Evaluation (TOE). The TOE is comprised of software-only. The software is comprised of the Universal Cisco Internet Operating System (IOS) software image Release IOS 15.5(2)T.



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:

- ESR Application Software which includes the VPN Client
- The following are considered to be in the IT Environment:
  - VPN Gateway
  - Linux OS
  - Hardware the ESR Software is installed

1.4 TOE Evaluated Configuration

Table 4: Evaluated Configurations

TOE	Cisco 5921 Embedded Services Router

The TOE consists of a software-only application. The underlying Operating System (OS) and hardware are not included in the TOE Boundary.

The TOE requires the following to run:

- Linux Kernel 2.6.32 or later version
- Superuser authority to run
- Minimum of 256 Mg memory
- x86 processors

1.5 Physical Scope of the TOE

The TOE is a software-only Linux application that makes up the Cisco 5921. The underlying operating system, hardware, and network, on which they reside is considered part of the environment.

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. Cryptographic Support
2. Full Residual Information Protection
3. Identification and Authentication
4. Security Management
5. Protection of the TSF
6. Trusted Channels

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

1.6.1 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 (see Table 5 for certificate references).

Table 5 FIPS References

Algorithm	Supported Mode	Algorithm Cert. #
		IOS
AES	CBC (128, 192, 256) CTR (256) GCM (128, 192, 256)	2785
Triple-DES	KO 1 & 2, CBC	1673
SHS (SHA-1, 256, and 512)	Byte Oriented	2340
HMAC SHA-1	Byte Oriented	1744
DRBG	CTR (using AES-256)	472
RSA	2048-3072 bit key	1457
ECDSA	P- 256, P- 384	486
CVL (as per SP 800-135)	IKEv1, IKEv2, SNMP, SSH, TLS	237

The TOE provides cryptography in support of IPsec connections. The cryptographic services provided by the TOE are described in Table 6 below.

Table 6: TOE Provided Cryptography

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

Cryptographic Method	Use within the TOE
AES	Used to encrypt IPsec session traffic.

The TOE can use X.509v3 certificates for authenticating IPsec sessions.

1.6.2 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.

1.6.3 Identification and authentication

The TOE performs device-level authentication of the remote device (IPsec peers or VPN Gateway). 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 TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for IPsec sessions.

1.6.4 Security Management

The TOE provides secure administrative services for management of general TOE configuration and the security functionality provided by the TOE. The TOE provides the ability to securely manage the TOE.

1.6.5 Protection of the TSF

The TOE protects against interference and tampering by untrusted subjects by providing a complete implementation of the Cisco IOS software. 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.

1.6.6 Trusted Channels

The TOE initiates IPsec tunnels with remote IPsec peers and VPN Gateways.

1.7 Excluded Functionality

The following functionality is excluded from the evaluation.

Table 7: 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.
Auditing	Auditing is not a baseline requirement and therefore not tested in the evaluated configuration.

Excluded Functionality	Exclusion Rationale
Network Time Protocol (NTP)	NTP is not a requirement on the TOE and therefore not tested in the evaluated configuration

The excluded functionality in table 7 will be disabled in the evaluated configuration. The exclusion of this functionality does not affect conformance to the Protection Profile for IPsec Virtual Private Network (VPN) Clients.

2 CONFORMANCE CLAIMS

2.1 Common Criteria Conformance Claim

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

The TOE and ST are 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 IPsec Virtual Private Network (VPN) Clients (VPNv1.4).

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

Table 8: Protection Profiles

Protection Profile	Version	Date
Protection Profile for IPsec Virtual Private Network (VPN) Clients	1.4	21 October 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 IPsec Virtual Private Network (VPN) Clients v1.4

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 VPNv1.4 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 VPNv1.4 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 VPNv1.4 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 VPNv1.4.

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 9 TOE Assumptions

Assumption	Assumption Definition
A.NO_TOE_BYPASS	Information cannot flow onto the network to which the VPN client's host is connected without passing through 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_CONFIG	Personnel configuring the TOE and its operational environment will follow the applicable security configuration guidance.

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 10 Threats

Threat	Threat Definition
T.TSF_CONFIGURATION	Failure to allow configuration of the TSF may prevent its users from being able to adequately implement their particular security policy, leading to a compromise of user information.
T.TSF_FAILURE	Security mechanisms of the TOE may fail, leading to a compromise of the TSF.
T.UNAUTHORIZED_ACCESS	A user may gain unauthorized access to the TOE data. 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 because it is not rendered inaccessible after it is done being used.

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 11 Security Objectives for the TOE

TOE Objective	TOE Security Objective Definition
O.VPN_TUNNEL	The TOE will provide a network communication channel protected by encryption that ensures that the VPN client communicates with an authenticated VPN gateway.
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.TOE_ADMINISTRATION	The TOE will provide mechanisms to allow administrators to be able to configure the TOE.
O.TSF_SELF_TEST	The TOE will provide the capability to test some subset of its security functionality to ensure it is operating properly.
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.

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 software. Thus, they will be satisfied largely through application of procedural or administrative measures.

Table 12 Security Objectives for the Environment

Environment Security Objective	IT Environment Security Objective Definition
OE.NO_TOE_BYPASS	Information cannot flow onto the network to which the VPN client's host is connected without passing through the TOE.
OE.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the operational environment.
OE.TRUSTED_CONFIG	Personnel configuring the TOE and its operational environment will follow the applicable security configuration guidance.

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 3, dated: July 2009* 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 VPNv1.4 itself, the formatting used in the VPNv1.4 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 <u>underlined</u> 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 13 Security Functional Requirements

Class Name	Component Identification	Component Name
FCS: Cryptographic support	FCS_CKM.1	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_RBG_EXT.1	Extended: Cryptographic Operation (Random Bit Generation)
FDP: User data protection	FDP_RIP.2	Full Residual Information Protection
FIA: Identification and authentication	FIA_PSK_EXT.1	Extended: Pre-Shared Key Composition
	FIA_X509_EXT.1	Extended: X.509 Certificates
FMT: Security management	FMT_SMF.1	Specification of Management Functions
FPT: Protection of the TSF	FPT_TST_EXT.1	TSF Testing

Class Name	Component Identification	Component Name
	FPT_TUD_EXT.1	Extended: Trusted Update
FTP: Trusted path/channels	FTP_ITC.1	Trusted Channel

5.3 SFRs Drawn from VPNv1.4

5.3.1 Cryptographic Support (FCS)

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

FCS_CKM.1.1(1) Refinement: The TOE 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 finite field-based key establishment schemes;*
- *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* (as defined in FIPS PUB 186-4, “Digital Signature Standard”)
- 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. See NIST Special Publication 800-57, “Recommendation for Key Management” for information about equivalent key strengths.

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

FCS_CKM.1.1(2) Refinement: The TOE shall generate **asymmetric** cryptographic keys **used for IKE peer authentication** in accordance with a:

[

- *FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.3 for RSA schemes;*

- *FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.4 for ECDSA schemes and implementing “NIST curves” P-256, P-384.*]

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

5.3.1.3 FCS_CKM_EXT.2 Cryptographic Key Storage

FCS_CKM_EXT.2.1 The TOE shall store persistent secrets and private keys when not in use in platform-provided key storage.

5.3.1.4 FCS_CKM_EXT.4 Cryptographic Key Zeroization

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

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

FCS_COP.1.1(1) Refinement: The TOE shall perform [*encryption and decryption*] in accordance with a specified cryptographic algorithm *AES operating in* **GCM and CBC mode** with cryptographic key sizes 128-bits and 256-bits that meets the following:

- **FIPS PUB 197, "Advanced Encryption Standard (AES)"**
- **NIST SP 800-38D, NIST SP 800-38A.**

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

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

- **FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3 for RSA scheme**
- **FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4 for ECDSA schemes and implementing "NIST curves" P-256, P-384**

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

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

FCS_COP.1.1(3) Refinement: The TOE shall perform [*cryptographic hashing services*] in accordance with a specified cryptographic algorithm **SHA-1, SHA-256, SHA-512 and message digest sizes 160, 256, 512 bits** that meet the following: *FIPS Pub 180-4, "Secure Hash Standard."*

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

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

5.3.1.9 **FCS_IPSEC_EXT.1 Explicit: IPSEC**

FCS_IPSEC_EXT.1.1 The TOE shall implement the IPsec architecture as specified in RFC 4301.

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

FCS_IPSEC_EXT.1.3 The TOE 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 TOE 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.

FCS_IPSEC_EXT.1.5 The TOE 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), 4307, and [no other RFCs for hash functions]].

FCS_IPSEC_EXT.1.6 The TOE 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 [no other algorithm].

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

FCS_IPSEC_EXT.1.8 The TOE shall ensure that [selection: IKEv2 SA lifetimes can be configured by an Administrator based on number of packets/number of bytes; 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/number of bytes; 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 TOE 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 TOE 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 TOE 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 TOE 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 TOE 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.

FCS_IPSEC_EXT.1.14 The TOE 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.

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

FCS_RBG_EXT.1.1 The TOE and TOE platform shall perform all deterministic random bit generation services in accordance with NIST Special Publication 800-90A using CTR_DRBG (AES).

FCS_RBG_EXT.1.2 The deterministic RBG shall be seeded by an entropy source that accumulates entropy from a software-based noise source with a minimum of 256 bits of entropy at least equal to the greatest security strength (according to NIST SP 800-57) of the keys and hashes that it will generate.

5.3.2 User data protection (FDP)

5.3.2.1 FDP_RIP.2 Full Residual Information Protection

FDP_RIP.2.1 The TOE shall enforce that any previous information content of a resource is made unavailable upon the allocation of the resource to all objects.

5.3.3 Identification and authentication (FIA)

5.3.3.1 FIA_PSK_EXT.1 Extended: Pre-Shared Key Composition

FIA_PSK_EXT.1.1 The TOE shall be able to use pre-shared keys for IPsec.

FIA_PSK_EXT.1.2 The TOE 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 TOE shall condition the text-based pre-shared keys by using [SHA-1] be able to accept bit-based pre-shared keys.

5.3.3.2 FIA_X509_EXT.1 Extended: X.509 Certificates

FIA_X509_EXT.1.1 The TOE shall validate certificates in accordance with the following rules:

- Perform RFC 5280 certificate validation and certificate path validation.
- Validate the revocation status of the certificate using the Online Certificate Status Protocol (OCSP) as specified in RFC 2560, a Certificate Revocation List (CRL) as specified in RFC 5759.
- Validate the certificate path by ensuring the basicConstraints extension is present and the cA flag is set to TRUE for all CA certificates.
- Validate the extendedKeyUsage field according to the following rules:
  - Certificates used for trusted updates, integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3).

FIA_X509_EXT.1.2 The TOE shall only treat a certificate as a CA certificate if the following is met: the basicConstraints extension is present and the cA flag is set to TRUE.

5.3.3.3 FIA_X509_EXT.2 Extended: X.509 Certificate Use and Management

FIA_X509_EXT.2.1 The TSF shall use X.509v3 certificates as defined by RFC 5280 to support authentication for IPsec exchanges, and no additional uses.

FIA_X509_EXT.2.2 When a connection to determine the validity of a certificate cannot be established, the TOE shall allow the administrator to choose whether to accept the certificate in these cases, not accept the certificate.

FIA_X509_EXT.2.3 The TOE shall not establish an SA if a certificate or certificate path is deemed invalid.

5.3.4 Security management (FMT)

5.3.4.1 FMT_SMF.1 Specification of Management Functions

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

- Configuration of IKE protocol version(s) used,
- Configure IKE authentication techniques used,
- Configure the cryptoperiod for the established session keys. The unit of measure for configuring the cryptoperiod shall be no greater than an hour,
- Configure certificate revocation check,
- Specify the algorithm suites that may be proposed and accepted during the IPsec exchanges,

- load X.509v3 certificates used by the security functions in this PP,
- ability to update the TOE, and to verify the updates,
- ability to configure all security management functions identified in other sections of this PP,
- specify VPN gateways to use for connections,
- specify client credentials to be used for connections,
- action to be taken when a connection to determine the validity of a certificate cannot be established, no other actions].

5.3.5 Protection of the TSF (FPT)

5.3.5.1 FPT_TST_EXT.1: TSF Testing

FPT_TST_EXT.1.1 The TOE 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 TOE 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)*.

5.3.5.2 FPT_TUD_(EXT).1 Extended: Trusted Update

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

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

FPT_TUD_(EXT).1.3 The TOE shall provide a means to verify firmware/software updates to the TOE using a digital signature mechanism and [no other function] prior to installing those updates.

5.3.6 Trusted Path/Channels (FTP)

5.3.6.1 FTP_ITC.1 Inter-TSF trusted channel

FTP_ITC.1.1 Refinement: The TOE shall **use IPsec** to provide a **trusted** communication channel between itself and **a VPN Gateway** 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 TOE shall permit *the TSF* to initiate communication via the trusted channel.

FTP_ ITC.1.3 The TOE shall initiate communication via the trusted channel *for all traffic traversing that connection.*

5.4 TOE SFR Dependencies Rationale for SFRs Found in VPNv1.4

The VPNv1.4 contains 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 3. The assurance requirements are summarized in the table below.

Table 14: 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 VPNv1.4. As such, the VPNv1.4 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 15: 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 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.
ALC_CMS.1	
ATE_IND.1	Cisco will provide the TOE for testing.
AVA_VAN.1	Cisco will provide the TOE for testing.

6 TOE SUMMARY SPECIFICATION

6.1 TOE Security Functional Requirement Measures

Table 16 identifies and describes how the Security Functional Requirements identified in section 5 of this ST are met by the TOE.

Table 16 How TOE SFRs Measures

TOE SFRs	How the SFR is Met
Security Functional Requirements Drawn from VPNv1.4	

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), for RSA key establishment schemes (conformant to NIST SP 800-56B).
	
	The TOE can create a RSA public-private key pair via the TOE's CLI 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.

FCS_CKM_EXT.2	The TOE stores all private keys in a secure directory that is not accessible to administrators. All pre-shared, private, and symmetric keys are stored in encrypted form using the TOE's AES encryption to additionally obscure access. Pre-shared keys can be used for authenticating to a remote IPsec peer or VPN Gateway. Symmetric keys are used for encrypting sensitive data. Private keys are the private key in the public-private asymmetric key pairs used for digital signature and decryption of data. 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.
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 17 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 186-4, “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 186-4, “Digital Signature Standard”.
FCS_COP.1(3)	The TOE provides cryptographic hashing services using SHA-1, SHA-256, and SHA-512 as specified in FIPS Pub 180-4 “Secure Hash Standard.”
FCS_COP.1(4)	The TOE provides keyed-hashing message authentication services using HMAC-

TOE SFRs	How the SFR is Met
	SHA-1 as specified in FIPS Pub 198-1,"The Keyed-Hash Message Authentication Code," and FIPS 180-4, "Secure Hash Standard." The block size produced for the HMAC-SHA1 is 160 bits.
FCS_IPSEC_EXT.1	The IPsec implementation provides both VPN peer-to-peer and TOE to VPN Gateway in both tunnel and transport mode. The VPN peer-to-peer tunnel allows for example the TOE and another VPN client to establish an IPsec tunnel to secure the passing of user data. The TOE to VPN Gateway configuration would be where the TOE connects into a remote VPN Gateway 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. Both OCSP 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. 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. 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. 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 using the configured isakmp policy. The key negotiated in phase 1 enables IKE peers to communicate securely in phase 2. During Phase 2, IKE establishes the IPsec SA using the configured IPsec transform-set. 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.

TOE SFRs	How the SFR is Met
	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. The TOE provides 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 less 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 negotiated in phase 1. 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. The administrator is instructed in the AGD to select a supported DH group. The TOE generates the secret value 'x' used in the IKEv2 Diffie-Hellman key exchange ('x' in g^x mod p) using the NIST approved DRBG specified in FCS_RBG_EXT.1 and having possible lengths of 256, 320, 384, 424, or 480 bits. When a random number is needed for a nonce, the probability that a specific nonce value will be repeated during the life a specific IPsec SA is less than 1 in 2^{128}. The nonce is likewise generated using the AES-CTR DRBG. 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 crypto map acl and does not match a non-crypto permit acl on the interface would be DISCARDED. Traffic that does not match a permit acl in the crypto map, but does match a non-crypto permit acl would be allowed to BYPASS the tunnel. For example, a non-crypto permit acl for icmp would allow ping traffic to flow unencrypted if a permit crypto map was not configured that matches the ping traffic. 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,

TOE SFRs	How the SFR is Met
	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_RBG_EXT.1	The TOE and underlying TOE hardware platform implements a NIST-approved AES-CTR Deterministic Random Bit Generator (DRBG), as specified in SP 800-90.
FDP_RIP.2	The TOE ensures that packets transmitted from the TOE do not contain residual information from previous packets. The packet data is zeroed out by the TOE software before it is passed to the Linux raw packet socket API. 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_PSK_EXT.1	Through the implementation of the CLI, 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 by the cryptographic module prior to use via SHA-1. The default SHA-1 algorithm can be changed through the crypto isakmp policy configuration and setting the hash algorithm.
FIA_X509_EXT.1 FIA_X509_EXT.2	The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for IPsec connections. Public key infrastructure (PKI) credentials, such as Rivest, Shamir, and Adelman (RSA) keys and certificates can be stored on the hard drive of the router. The TOE maintains the key pair and associates it with the X.509.v3 certificate used for IPsec. The certificates themselves provide protection in that they are digitally signed. If a certificate is modified in any way, it would be invalidated. The validity of the certificate and the certificate chain is verified by the TOE. 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 IPsec credentials for deployment can be implemented using the USB tokens. Both OCSP 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. Certificates used for trusted updates and integrity verification will have the code signing purpose object identifier (id-kp 3 with OID 1.3.6.1.5.5.7.3.3). The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for IPsec connections. The certificates themselves provide protection in that they are digitally signed. When a connection to determine the validity of a certificate cannot be established, then the TOE allows the administrator to accept or reject the certificate via the CLI.
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 SSH, a terminal server, or at the local console. The use of SSH or a terminal server to connect to the CLI was not specifically tested in the evaluation. The focus of this test is to demonstrate the functionality as performed through the CLI which was demonstrated via the local console.

TOE SFRs	How the SFR is Met
	Refer to the Guidance documentation for configuration syntax, commands, and information related to each of these functions. All of these functions can be performed via the CLI either locally or remotely. The specific management capabilities available from the TOE include: • Configuration of IKE protocol version(s) used, • Configure IKE authentication techniques used, • Configure the session key lifetimes of no greater than an hour, • Configure certificate revocation check, • Specify the algorithm suites that may be proposed and accepted during the IPsec exchanges, • load X.509v3 certificates used by the security functions in this ST, • ability to update the TOE, and to verify the updates, • ability to configure all security management functions identified in other sections of this ST, • specify VPN gateways to use for connections, • specify client credentials to be used for connections, • accept or deny the validity of the certificate. The TOE provides the ability for Authorized Administrators to configure the VPN gateways the VPN client will connect to. The client credentials can be a client X.509 certificate and/or pre-shared key that are used for authentication to the VPN Gateway.
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.
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

TOE SFRs	How the SFR is Met
	functionality. Software images are available from Cisco.com at the following: http://www.cisco.com/cisco/software/navigator.html 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 administrator, after loading the image onto the device, needs to perform a verify operation on the mage to confirm it is valid. The TOE will identify if the image is valid or not, and then the administrator will manually reject a bad image and does not proceed with the installation. The certificate issued to the TOE used for digital signature verification 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 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.
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.

6.2 Key Zeroization

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

Table 17: 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.

Name	Description	Zeroization
		Overwrtten with: 0x00
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

7 ANNEX A: REFERENCES

The following documentation was used to prepare this ST:

Table 18: References

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