# Cisco 900 Series Integrated Services Routers (ISR) running IOS v15.9 Security Target

**Version:** 0.19
**Date:** 24 October 2022

Cisco Systems, Inc.

# Table of Contents

# List of Tables



# List of Figures

## *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 |
| 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 |
| ESP | Encapsulating Security Payload |
| ESPr | Embedded Services Processors |
| GE | Gigabit Ethernet port |
| HTTPS | Hyper-Text Transport Protocol Secure |
| IT | Information Technology |
| NDcPP | collaborative Protection Profile for Network Devices |
| OS | Operating System |
| PoE | Power over Ethernet |
| PP | Protection Profile |
| SA | Security Association |
| SFP | Small–form-factor pluggable port |
| SHS | Secure Hash Standard |
| ST | Security Target |
| TCP | Transport Control Protocol |
| TSC | TSF Scope of Control |
| TSF | TOE Security Function |
| TSP | TOE Security Policy |
| WAN | Wide Area Network |
| WIC | WAN Interface Card |

# Terminology

The following terms are common and may be used in this Security Target.

**Table 2 Terminology**

| Term | Definition |
|---|---|
| Authorized Administrator | Any user who has been assigned to a privilege level that is permitted to perform all TSF-related functions. |
| Peer | Another router on the network that the TOE interfaces with. |
| Remote VPN Peer | A remote VPN Peer is another network device that the TOE sets up a VPN connection with. This could be a VPN client or another router. |
| Security Administrator | Synonymous with Authorized Administrator for the purposes of this evaluation. |
| User | Any entity (human user or external IT entity) outside the TOE that interacts with the TOE. |
| Vty | vty is a term used by Cisco to describe a single terminal (whereas Terminal is more of a verb or general action term). For configuration purposes vty defines the line for remote access policies to the router. |

# 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), Cisco 900 Series Integrated Services Routers (ISR) running IOS v15.9. 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.

# 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]
- Key Zeroization [Section 7]
- Annex A: References
- Annex B: Technical Decisions

The structure and content of this ST comply with the requirements specified in the Common Criteria (CC), Part 1, Annex A, and Part 3, Chapter 11.

## 1.1 ST and TOE Reference

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

**Table 3 ST and TOE Identification**

| Name | Description | Delivery Method |
|---|---|---|
| ST Title | *Cisco 900 Series Integrated Services Routers (ISR) running IOS v15.9 Security Target* | *https://www.niap-ccevs.org/Product/index.cfm* |
| ST Version | 0.19 | |
| Publication Date | 24 October 2022 | |
| Vendor and ST Author | Cisco Systems, Inc. | |
| TOE Reference | Cisco 900 Series Integrated Services Routers (ISR) running IOS v15.9 | |
| TOE Hardware Models | C921-4P, C921J-4P, C926-4P, C927-4P, and C931-4P | Delivered through courier delivery |
| TOE Software Version | IOS v15.9 (build number v15.9.3M4) | Download at www.cisco.com |
| TOE documentation[1] | Installation, configuration, and administration guides to properly manage the TOE | Download at www.cisco.com |
| Keywords | Router, Network Appliance, Data Protection, Authentication, Cryptography, Secure Administration, Network Device, Virtual Private Network(VPN), VPN Gateway | |

[1] A detailed list of documentation needed to install, configure, and administer the TOE is found in the *Cisco 900 Series Integrated Services Routers (ISR) Common Criteria Operational User Guidance And Preparative Procedures*

## 1.2 TOE Overview

The Cisco Integrated Services Router (ISR) (herein after referred to as the ISR900) is a purpose-built, routing platform that includes VPN functionality.

Cisco IOS software is a Cisco -developed highly configurable proprietary operating system that provides for efficient and effective switching and routing. Although IOS performs many networking functions, this Security Target only addresses the functions that provide for the security of the TOE itself.

## 1.3 TOE Product Type

The TOE is a network device that includes VPN functionality as defined in collaborative Protection Profile for Network Devices v2.2e (NDcPP v2.2e) and PP-Module for Virtual Private Network (VPN) Gateways v1.1 (MOD_VPNGW v1.1).

The ISR900 routers combine WAN, switching, security, and advanced connectivity options in a compact, fanless platform. The ISR900 routers allows customers to create a highly secure, enterprise-class network. The ISR900 router is well suited for deployment as Customer Premises Equipment (CPE) in enterprise small branch offices and in service provider managed-service environments.

## 1.4 Supported non-TOE Hardware/ Software/ Firmware

The TOE supports the following hardware, software, and firmware in its environment when the TOE is configured in its evaluated configuration:

**Table 4 IT Environment Components**

| Component | Required | Usage/Purpose Description for TOE performance |
|---|---|---|
| RADIUS AAA Server | Yes | This includes any IT environment RADIUS AAA server that provides single-use authentication mechanisms. This can be any RADIUS AAA server that provides single-use authentication. The TOE correctly leverages the services provided by this RADIUS AAA server to provide single-use authentication to administrators. |
| Management Workstation with SSH Client | Yes | This includes any IT Environment Management workstation with an 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. |
| Local Console | Yes | 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. |
| Certification Authority (CA) | Yes | 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. |
| Remote VPN Peer | Yes | This includes any VPN Peer (Gateway, Endpoint, another instance of the TOE) with which the TOE participates in VPN communications. Remote VPN Peers may be any device that supports IPsec VPN communications. Another instance of the TOE used as a VPN Peer would be installed in the evaluated configuration, and likely administered by the same personnel. |
| Audit (syslog) Server | Yes | This includes any syslog server to which the TOE would transmit syslog messages. Also referred to as audit server in the ST |

## 1.5 TOE Description

This section provides an overview of the ISR900 Target of Evaluation (TOE). This section also defines the TOE components included in the evaluated configuration of the TOE. The TOE is comprised of both software and hardware. The hardware models included in the evaluation are: C921-4P, C921J-4P, C926-4P, C927-4P, and C931-4P. The software is comprised of the Cisco IOS 15.9.

The ISR900 consists of the following architectural features:

- Default and maximum DRAM - Default 1 GB
- Default and maximum flash memory - 2 GB on all Cisco ISR900 models
- Separate console ports - RJ-45
- USB 2.0 - One USB 2.0 Type A port
- WAN
  - C921-4P, C921J-4P, C931-4P - 2 ports Gigabit Ethernet (GE)
  - C926-4P, C927-4P - 1 port Gigabit Ethernet (GE)
- LAN switch - 4 Layer 2 GE LAN ports

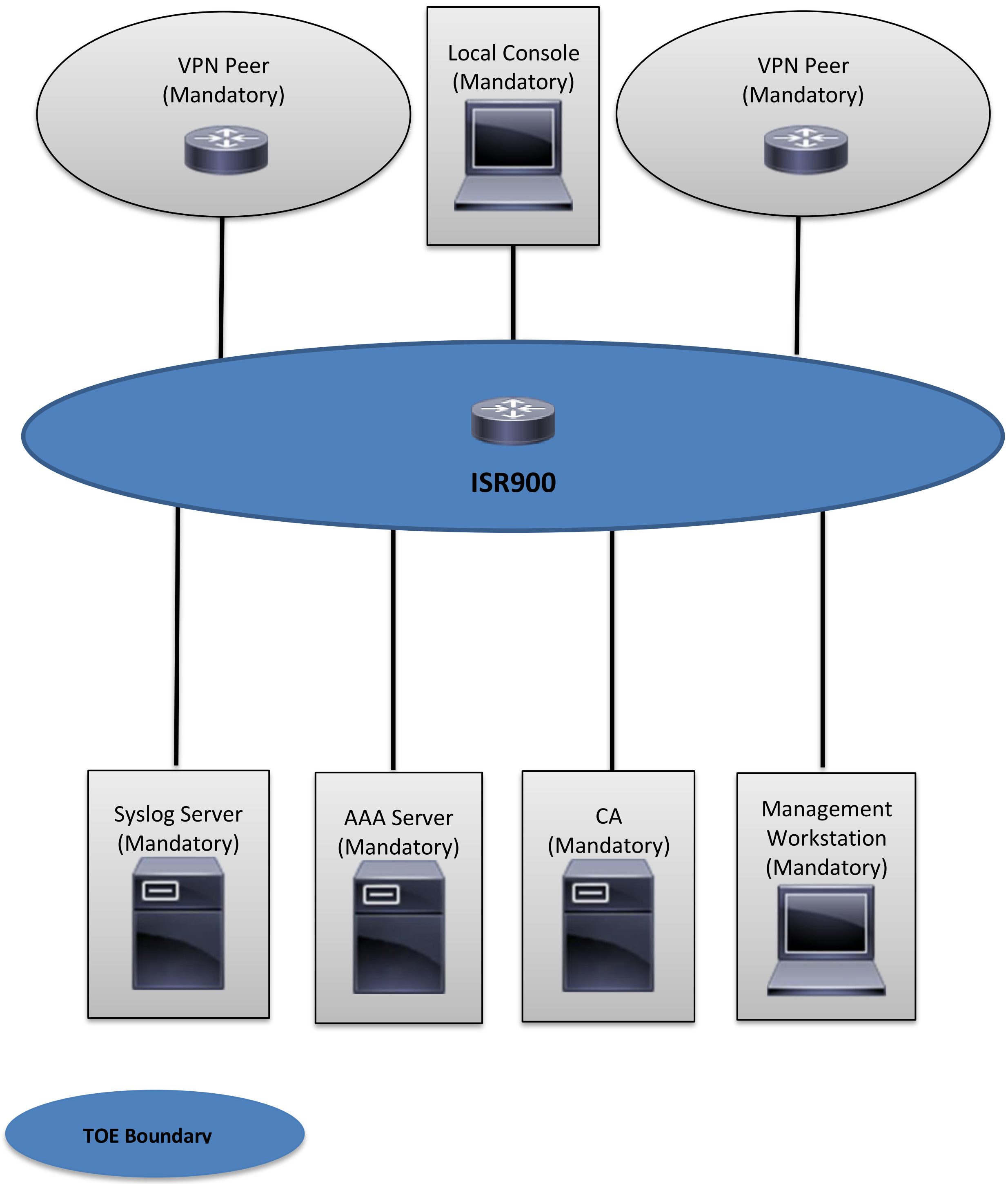
Cisco IOS is a Cisco -developed highly configurable proprietary operating system that provides for efficient and effective routing and switching. Although IOS performs many networking functions, this TOE only addresses the functions that provide for the security of the TOE itself as described in Section 1.8 Logical Scope of the TOE below.

## 1.6 TOE Evaluated Configuration

The TOE consists of one physical device as specified in section 1.7 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. NVRAM and flash provide local storage.

The following figure provides a visual depiction of an example TOE deployment:

**Figure 1 TOE Example Deployment**



**TOE Boundary**

The previous figure includes the following:

- The following are considered to be in the IT Environment:
    - VPN Peers (secure connection is IPsec)
    - Management Workstation (secure connection is SSHv2)
    - RADIUS AAA (Authentication) Server (secure connection is IPsec)
    - Audit (Syslog) Server (secure connection is IPsec)
    - Local Console (directly connected)
    - Certification Authority (CA) (secure connection is IPsec)

NOTE: While the previous figure includes several non-TOE IT environment devices, the TOE is only the ISR900 device. Only one TOE device is required for deployment in an evaluated configuration.

## 1.7 Physical Scope of the TOE

The TOE is a hardware and software solution that makes up the router models as follows:

C921-4P, C921J-4P, C926-4P, C927-4P, and C931-4P

The network, on which they reside, is considered part of the environment. The software is pre-installed and is comprised of the Cisco IOS software image Release 15.9. In addition, the software image is also downloadable from the Cisco web site. A login id and password are required to download the software image. The TOE is comprised of the following physical specifications as described in Table 5 below:

**Table 5 Hardware Models and Specifications**

| Hardware | Picture | Features |
|---|---|---|
| Cisco ISR900<br><br>C921-4P, C921J-4P, C926-4P, C927-4P, and C931-4P | C921-4P, C921J-4P, and C931-4P:<br><br>C926-4P and C927-4P:<br> | **Physical dimensions** (H x W x D)<br>• C921-4P, C921J-4P, and C931-4P: 1.70 x 9.0 x 9.5 in. (4.32 x 22.86 x 24.13 cm)<br>• C926-4P and C927-4P: 1.10 x 10.20 x 7.00 in. (2.80 x 25.91 x 17.78 cm)<br><br>**Processor:**<br>Intel Atom C3558 (Goldmont)<br><br>**Memory**<br>• Default and maximum DRAM - Default 1 GB<br>• Default and maximum flash memory - 2 GB<br><br>**Interfaces**<br>• Separate console ports - RJ-45<br>• USB 2.0 - One USB 2.0 Type A port<br>• WAN<br>- C921-4P, C921J-4P, and C931-4P - 2 ports Gigabit Ethernet (GE)<br>- C926-4P and C927-4P - 1 port Gigabit Ethernet (GE)<br>• LAN switch - 4 Layer 2 GE LAN ports<br><br>**Power**<br>• AC input voltage: Universal 100 to 240 VAC |

For ordering of the TOE hardware and delivery via commercial carriers, visit Cisco.com - Cisco 900 Series Integrated Services Routers .

The software is the Cisco IOS software image Release 15.9. For ordering and downloading the TOE software, see Cisco Software Central.

The TOE guidance documentation that is also considered to be part of the TOE is the *Cisco 900 Series Integrated Services Routers (ISR) running IOS v15.9 Common Criteria Operational User Guidance and Preparative Procedures*. This document is downloadable from the http://cisco.com web site at:

https://www.cisco.com/c/en/us/solutions/industries/government/global-government-certifications/common-criteria.html

In Table 1 Common Criteria Certified Product Guidance, enter the certified product name or simply click on the certification date for the product. A PDF version of the document will be displayed, which can be downloaded and saved.

## 1.8 Logical Scope of the TOE

The TOE is comprised of the following security features:

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

These features are described in more detail in the subsections below. In addition, the TOE implements all RFCs of the NDcPP v2.2e and MOD_VPNGW v1.1 as necessary to satisfy testing/assurance measures prescribed therein.

### 1.8.1 Security Audit

The TOE provides extensive auditing capabilities. The TOE can audit events related to cryptographic functionality, identification and authentication, and administrative actions. The TOE generates an audit record for each auditable event. Each security relevant audit event has the date, timestamp, event description, and subject identity. The administrator configures auditable events, performs back-up operations, and manages audit data storage. The TOE provides the administrator with a circular audit trail. The TOE is configured to transmit its audit messages to an external syslog server over an encrypted channel.

### 1.8.2 Cryptographic Support

The TOE provides cryptography in support of other TOE security functionality. All the algorithms claimed have CAVP certificates (Operational Environment – Intel Atom C3558). The TOE leverages the IOS Common Cryptographic Module (IC2M) Rel5 (see Table 6 for certificate references).

**Table 6 FIPS References**

| Algorithm | Description | Supported Mode | Module | CAVP Cert. # | SFR |
|---|---|---|---|---|---|
| AES | Used for symmetric encryption/decryption | CBC (128, 192, and 256)<br>GCM (128, 192, and 256) | IC2M | C1802 | FCS_COP.1/DataEncryption<br>FCS_IPSEC_EXT.1 |

| Algorithm | Description | Supported Mode | Module | CAVP Cert. # | SFR |
|---|---|---|---|---|---|
| SHS (SHA-1, SHA-256, SHA-384 and SHA-512) | Cryptographic hashing services | Byte Oriented | IC2M | C1802 | FCS_COP.1/Hash |
| HMAC (HMAC-SHA-1, SHA-256, SHA-512) | Keyed hashing services and digital signature | Byte Oriented | IC2M | C1802 | FCS_COP.1/KeyedHash<br>FCS_IPSEC_EXT.1 |
| DRBG | Deterministic random bit generation services in accordance with ISO 18031:2011 | CTR_DRBG (AES 256) | IC2M | C1802 | FCS_RBG_EXT.1 |
| RSA | Signature Verification and key transport | PKCS#1 v.1.5, 3072 bit key, FIPS 186-4 Key Gen | IC2M | C1802 | FCS_CKM.1<br>FCS_COP.1/SigGen |
| ECDSA | Cryptographic Signature services | FIPS 186-4, Digital Signature Standard (DSS) | IC2M | C1802 | FCS_CKM.1<br>FCS_COP.1/SigGen |
| CVL-KAS-ECC | Key Agreement | NIST Special Publication 800-56A | IC2M | C1802 | FCS_CKM.2 |

The TOE provides cryptography in support of VPN connections and remote administrative management via SSHv2 and IPsec to secure the transmission of audit records to the remote syslog server. In addition, IPsec is used to secure the session between the TOE and the authentication servers as well as to protect communications with a CA.

The cryptographic services provided by the TOE are described in Table 7 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.<br>Used in SSH session establishment.<br>X.509 certificate signing |
| SP 800-90 RBG | Used in IPsec session establishment.<br>Used in SSH session establishment. |
| SHS | Used to provide IPsec traffic integrity verification<br>Used to provide SSH traffic integrity verification<br>Used for keyed-hash message authentication |

| Cryptographic Method | Use within the TOE |
|---|---|
| AES | Used to encrypt IPsec session traffic.<br>Used to encrypt SSH session traffic. |
| RSA | Used in IKE protocols peer authentication<br>Used to provide cryptographic signature services |
| ECDSA | Used to provide cryptographic signature services<br>Used in Cryptographic Key Generation |
| FFC DH | Used as the Key exchange method for SSH and IPsec |
| ECC DH | Used as the Key exchange method for IPsec |

### 1.8.3 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 TOE supports use of IKEv2 pre-shared keys for authentication of IPsec tunnels. 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 TOE's 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. 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 supports the use of a RADIUS AAA server (part of the IT Environment) for authentication of administrative users attempting to connect to the TOE's CLI.
The TOE provides an automatic lockout when a user attempts to authenticate and enters invalid information. After a defined number of authentication attempts exceeding the configured allowable attempts, the user is locked out until an authorized administrator can enable the user account.

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

### 1.8.4 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:

- Administration of the TOE locally and remotely;
- 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 verification of the updates;
- Configuration of IPsec functionality.

The TOE supports two separate administrator roles: non-privileged administrator and privileged administrator. Only the privileged administrator can perform the above security relevant management functions. Management of the TSF data is restricted to Security Administrators. The ability to enable, disable, determine, and modify the behavior of all of the security functions of the TOE is restricted to authorized administrators.

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.

### 1.8.5 Packet Filtering

The TOE provides packet filtering and secure IPsec tunneling. The tunnels can be established between two trusted VPN peers 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.

### 1.8.6 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.

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. Finally, the TOE performs testing to verify correct operation of the router itself and that of the cryptographic module.

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

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 information to or from the TOE and reloads.

### 1.8.7 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. Sessions can also be terminated if an Authorized Administrator enters the “exit” or “logout” command.

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

### 1.8.8 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 IPsec sessions. The peer-to-peer IPsec sessions are also used for securing the communications between the TOE and authentication server, as well as to protect communications with a CA.

## 1.9 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 | This mode of operation includes non-FIPS allowed operations. |
| Hypertext Transfer Protocol (HTTP) | HTTP Is not associated with Security Functional Requirements claimed in NDcPP. Use tunnelling through IPSEC. |
| Hypertext Transfer Protocol Secure (HTTPS) | HTTPS is not associated with Security Functional Requirements claimed in NDcPP. Use tunnelling through IPSEC. |
| SSH (Client Side) | The TOE acts as an SSH server. It does not operate as an SSH client so no Security Functional Requirements are claimed from the NDcPP. |
| Telnet | Telnet sends authentication data in plain text. This feature must remain disabled in the evaluated configuration. SSHv2 must be used to secure the trusted path for remote administration for all SSHv2 sessions. |
| Transport Layer Security (TLS) | TLS is not associated with Security Functional Requirements claimed in [NDcPP] IPsec is used instead. |

These services will be disabled by configuration settings as described in the Guidance documents (AGD). The exclusion of this functionality does not affect compliance to the NDcPP v2.2e and MOD_VPNGW v1.1.

# 2 Conformance Claims

## 2.1 Common Criteria Conformance Claim

The TOE and ST are compliant with the Common Criteria (CC) Version 3.1, Revision 5, dated: April 2017. 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 Profiles as listed in Table 9 Protection Profiles below:

**Table 9 Protection Profiles**

| Protection Profile | Version | Date |
|---|---|---|
| collaborative Protection Profile for Network Devices | 2.2e | 23 March 2020 |
| PP-Module for Virtual Private Network (VPN) Gateways | 1.1 | 18 June 2020 |

## 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 collaborative Protection Profiles:

- collaborative Protection Profile for Network Devices (NDcPP) v2.2e
- PP-Module for Virtual Private Network (VPN) Gateways (MOD_VPNGW) v1.1

### 2.3.2 TOE Security Problem Definition Consistency

The Assumptions, Threats, and Organizational Security Policies included in the Security Target represent the Assumptions, Threats, and Organizational Security Policies specified in the NDcPP v2.2e and MOD_VPNGW v1.1 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 NDcPP v2.2e and MOD_VPNGW v1.1 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 NDcPP v2.2e and MOD_VPNGW v1.1 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 the NDcPP v2.2e and MOD_VPNGW v1.1.

### 2.3.4 Statement of Extended Components Definitions Consistency

The Extended Components Security Functional Requirements included in the Security Target represent the Security Functional Requirements specified in the NDcPP v2.2e and MOD_VPNGW v1.1 for which conformance is claimed verbatim. All concepts covered in the Protection Profile's Statement of Extended Security Requirements are included in this Security Target.

# 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. Organizational Security Policies (OSPs) are identified as P.osp with "osp" 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 |
|---|---|
| A.PHYSICAL_PROTECTION | The Network Device is assumed to be physically protected in its operational environment and not subject to physical attacks that compromise the security and/or interfere with the device's physical interconnections and correct operation. This protection is assumed to be sufficient to protect the device and the data it contains. As a result, the cPP does not include any requirements on physical tamper protection or other physical attack mitigations. The cPP does not expect the product to defend against physical access to the device that allows unauthorized entities to extract data, bypass other controls, or otherwise manipulate the device. |
| A.LIMITED_FUNCTIONALITY | The device is assumed to provide networking functionality as its core function and not provide functionality/ services that could be deemed as general purpose computing. For example, the device should not provide a computing platform for general purpose applications (unrelated to networking functionality). |
| A.NO_THRU_TRAFFIC_PROTECTION | A standard/generic Network Device does not provide any assurance regarding the protection of traffic that traverses it. The intent is for the Network Device to protect data that originates on or is destined to the device itself, to include administrative data and audit data. Traffic that is traversing the Network Device, destined for another network entity, is not covered by the ND cPP. It is assumed that this protection will be covered by cPPs and PP-Modules for particular types of Network Devices (e.g., firewall). |
| A.TRUSTED_ADMINISTRATOR | The Security Administrator(s) for the Network Device are assumed to be trusted and to act in the best interest of security for the organization. This includes appropriately trained, following policy, and adhering to guidance documentation. Administrators are trusted to ensure passwords/credentials have sufficient strength and entropy and to lack malicious intent when administering the device. The Network Device is not expected to be capable of defending against a malicious Administrator that actively works to bypass or compromise the security of the device. |

| Assumption | Assumption Definition |
|---|---|
| | For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are expected to fully validate (e.g., offline verification) any CA certificate (root CA certificate or intermediate CA certificate) loaded into the TOE's trust store (aka 'root store', ' trusted CA Key Store', or similar) as a trust anchor prior to use (e.g., offline verification). |
| A.REGULAR_UPDATES | The Network Device firmware and software is assumed to be updated by an Administrator on a regular basis in response to the release of product updates due to known vulnerabilities. |
| A.ADMIN_CREDENTIALS_SECURE | The Administrator's credentials (private key) used to access the Network Device are protected by the platform on which they reside. |
| A.RESIDUAL_INFORMATION | The Administrator must ensure that there is no unauthorized access possible for sensitive residual information (e.g., cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment. |
| 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 |
|---|---|
| T.UNAUTHORIZED_ADMINISTRATOR_ACCESS | Threat agents may attempt to gain Administrator access to the Network Device by nefarious means such as masquerading as an Administrator to the device, masquerading as the device to an Administrator, replaying an administrative session (in its entirety, or selected portions), or performing man-in-the-middle attacks, which would provide access to the administrative session, or sessions between Network Devices. Successfully gaining Administrator access allows malicious actions that compromise the security functionality of the device and the network on which it resides. |
| T.WEAK_CRYPTOGRAPHY | Threat agents may exploit weak cryptographic algorithms or perform a cryptographic exhaust against the key space. Poorly chosen encryption algorithms, modes, and key sizes will allow attackers to compromise the algorithms, or brute force exhaust the key space and give them unauthorized access allowing them to read, manipulate and/or control the traffic with minimal effort. |
| T.UNTRUSTED_COMMUNICATION_CHANNELS | Threat agents may attempt to target Network Devices that do not use standardized secure tunneling protocols to protect the critical network traffic. Attackers may take advantage of poorly designed protocols or poor key management to successfully perform man-in-the-middle attacks, replay attacks, etc. Successful attacks will result in loss of confidentiality and integrity of the critical network traffic, and potentially could lead to a compromise of the Network Device itself. |

| Threat | Threat Definition |
|---|---|
| T.WEAK_AUTHENTICATION_ENDPOINTS | Threat agents may take advantage of secure protocols that use weak methods to authenticate the endpoints, e.g., ashared password that is guessable or transported as plaintext. The consequences are the same as a poorly designed protocol, the attacker could masquerade as the Administrator or another device, and the attacker could insert themselves into the network stream and perform a man-in-the-middle attack. The result is the critical network traffic is exposed and there could be a loss of confidentiality and integrity, and potentially the Network Device itself could be compromised. |
| T.UPDATE_COMPROMISE | Threat agents may attempt to provide a compromised update of the software or firmware which undermines the security functionality of the device. Non-validated updates or updates validated using non-secure or weak cryptography leave the update firmware vulnerable to surreptitious alteration. |
| T.UNDETECTED_ACTIVITY | Threat agents may attempt to access, change, and/or modify the security functionality of the Network Device without Administrator awareness. This could result in the attacker finding an avenue (e.g., misconfiguration, flaw in the product) to compromise the device and the Administrator would have no knowledge that the device has been compromised. |
| T.SECURITY_FUNCTIONALITY_COMPROMISE | Threat agents may compromise credentials and device data enabling continued access to the Network Device and its critical data. The compromise of credentials includes replacing existing credentials with an attacker's credentials, modifying existing credentials, or obtaining the Administrator or device credentials for use by the attacker. |
| T.PASSWORD_CRACKING | Threat agents may be able to take advantage of weak administrative passwords to gain privileged access to the device. Having privileged access to the device provides the attacker unfettered access to the network traffic and may allow them to take advantage of any trust relationships with other Network Devices. |
| T.SECURITY_FUNCTIONALITY_FAILURE | An external, unauthorized entity could make use of failed or compromised security functionality and might therefore subsequently use or abuse security functions without prior authentication to access, change or modify device data, critical network traffic or security functionality of the device. |

| Threat | Threat Definition |
|---|---|
| T.NETWORK_DISCLOSURE | Devices on a protected network may be exposed to threats presented by devices located outside the protected network, which may attempt to conduct unauthorized activities. If known malicious external devices are able to communicate with devices on the protected network, or if devices on the protected network can establish communications with those external devices (e.g., as a result of a phishing episode or by inadvertent responses to email messages), then those internal devices may be susceptible to the unauthorized disclosure of information.<br><br>From an infiltration perspective, VPN gateways serve not only to limit access to only specific destination network addresses and ports within a protected network, but whether network traffic will be encrypted or transmitted in plaintext. With these limits, general network port scanning can be prevented from reaching protected networks or machines, and access to information on a protected network can be limited to that obtainable from specifically configured ports on identified network nodes (e.g., web pages from a designated corporate web server). Additionally, access can be limited to only specific source addresses and ports so that specific networks or network nodes can be blocked from accessing a protected network thereby further limiting the potential disclosure of information.<br><br>From an exfiltration perspective, VPN gateways serve to limit how network nodes operating on a protected network can connect to and communicate with other networks limiting how and where they can disseminate information. Specific external networks can be blocked altogether, or egress could be limited to specific addresses and/or ports. Alternately, egress options available to network nodes on a protected network can be carefully managed in order to, for example, ensure that outgoing connections are encrypted to further mitigate inappropriate disclosure of data through packet sniffing. |

| Threat | Threat Definition |
|---|---|
| T.NETWORK_ACCESS | Devices located outside the protected network may seek to exercise services located on the protected network that are intended to only be accessed from inside the protected network or only accessed by entities using an authenticated path into the protected network. Devices located outside the protected network may, likewise, offer services that are inappropriate for access from within the protected network.<br><br>From an ingress perspective, VPN gateways can be configured so that only those network servers intended for external consumption by entities operating on a trusted network (e.g., machines operating on a network where the peer VPN gateways are supporting the connection) are accessible and only via the intended ports. This serves to mitigate the potential for network entities outside a protected network to access network servers or services intended only for consumption or access inside a protected network.<br><br>From an egress perspective, VPN gateways can be configured so that only specific external services (e.g., based on destination port) can be accessed from within a protected network, or moreover are accessed via an encrypted channel. For example, access to external mail services can be blocked to enforce corporate policies against accessing uncontrolled e-mail servers, or, that access to the mail server must be done over an encrypted link. |
| T.NETWORK_MISUSE | Devices located outside the protected network, while permitted to access particular public services offered inside the protected network, may attempt to conduct inappropriate activities while communicating with those allowed public services. Certain services offered from within a protected network may also represent a risk when accessed from outside the protected network.<br><br>From an ingress perspective, it is generally assumed that entities operating on external networks are not bound by the use policies for a given protected network. Nonetheless, VPN gateways can log policy violations that might indicate violation of publicized usage statements for publicly available services.<br><br>From an egress perspective, VPN gateways can be configured to help enforce and monitor protected network use policies. As explained in the other threats, a VPN gateway can serve to limit dissemination of data, access to external servers, and even disruption of services – all of these could be related to the use policies of a protected network and as such are subject in some regards to enforcement. Additionally, VPN gateways can be configured to log network usages that cross between protected and external networks and as a result can serve to identify potential usage policy violations. |

| Threat | Threat Definition |
|---|---|
| T.REPLAY_ATTACK | If an unauthorized individual successfully gains access to the system, the adversary may have the opportunity to conduct a "replay" attack. This method of attack allows the individual to capture packets traversing throughout the network and send the packets at a later time, possibly unknown by the intended receiver. Traffic is subject to replay if it meets the following conditions:<br>• Cleartext: an attacker with the ability to view unencrypted traffic can identify an appropriate segment of the communications to replay as well in order to cause the desired outcome.<br>• No integrity: alongside cleartext traffic, an attacker can make arbitrary modifications to captured traffic and replay it to cause the desired outcome if the recipient has no means to detect these modifications. |
| T.DATA_INTEGRITY | Devices on a protected network may be exposed to threats presented by devices located outside the protected network, which may attempt to modify the data without authorization. If known malicious external devices are able to communicate with devices on the protected network or if devices on the protected network can establish communications with those external devices, then the data contained within the communications may be susceptible to a 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 Problem Definition

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 |
|---|---|
| O.ADDRESS_FILTERING | To address the issues associated with unauthorized disclosure of information, inappropriate access to services, misuse of services, disruption or denial of services, and network-based reconnaissance, compliant TOE's will implement Packet Filtering capability. That capability will restrict the flow of network traffic between protected networks and other attached networks based on network addresses of the network nodes originating (source) and/or receiving (destination) applicable network traffic as well as on established connection information. |
| O.AUTHENTICATION | To further address the issues associated with unauthorized disclosure of information, a compliant TOE's authentication ability (IPsec) will allow a VPN peer to establish VPN connectivity with another VPN peer and ensure that any such connection attempt is both authenticated and authorized. VPN endpoints authenticate each other to ensure they are communicating with an authorized external IT entity. |
| O.CRYPTOGRAPHIC_FUNCTIONS | To address the issues associated with unauthorized disclosure of information, inappropriate access to services, misuse of services, disruption of services, and network-based reconnaissance, compliant TOE's will implement a cryptographic capabilities. These capabilities are intended to maintain confidentiality and allow for detection and modification of data that is transmitted outside of the TOE. |
| O.FAIL_SECURE | There may be instances where the TOE's hardware malfunctions or the integrity of the TOE's software is compromised, the latter being due to malicious or non-malicious intent. To address the concern of the TOE operating outside of its hardware or software specification, the TOE will shut down upon discovery of a problem reported via the self-test mechanism and provide signature-based validation of updates to the TSF. |
| O.PORT_FILTERING | To further address the issues associated with unauthorized disclosure of information, etc., a compliant TOE's port filtering capability will restrict the flow of network traffic between protected networks and other attached networks based on the originating (source) and/or receiving (destination) port (or service) identified in the network traffic as well as on established connection information. |
| O.SYSTEM_MONITORING | To address the issues of administrators being able to monitor the operations of the VPN gateway, it is necessary to provide a capability |

| TOE Objective | TOE Security Objective Definition |
|---|---|
| | to monitor system activity. Compliant TOEs will implement the ability to log the flow of network traffic. Specifically, the TOE will provide the means for administrators to configure packet filtering rules to 'log' when network traffic is found to match the configured rule. As a result, matching a rule configured to 'log' will result in informative event logs whenever a match occurs. In addition, the establishment of security associations (SAs) is auditable, not only between peer VPN gateways, but also with certification authorities (CAs). |
| O.TOE_ADMINISTRATION | TOEs will provide the functions necessary for an administrator to configure the packet filtering rules, as well as the cryptographic aspects of the IPsec protocol that are enforced by the TOE. |

## 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 |
|---|---|
| OE.PHYSICAL | Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment. |
| 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.NO_THRU_TRAFFIC_PROTECTION | The TOE does not provide any protection of traffic that traverses it. It is assumed that protection of this traffic will be covered by other security and assurance measures in the operational environment. |
| OE.TRUSTED_ADMIN | Security Administrators are trusted to follow and apply all guidance documentation in a trusted manner.<br><br>For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are assumed to monitor the revocation status of all certificates in the TOE's trust store and to remove any certificate from the TOE's trust store in case such certificate can no longer be trusted. |
| OE.UPDATES | The TOE firmware and software is updated by an Administrator on a regular basis in response to the release of product updates due to known vulnerabilities. |
| OE.ADMIN_CREDENTIALS_SECURE | The Administrator's credentials (private key) used to access the TOE must be protected on any other platform on which they reside. |
| OE.RESIDUAL_INFORMATION | The Security Administrator ensures that there is no unauthorized access possible for sensitive residual information (e.g., cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment. For vNDs, this applies when the physical platform on which the VM runs is removed from its operational environment. |
| OE.CONNECTIONS | 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. |

# 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 5, dated: April 2017* 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:

- Assignment: Indicated with *italicized* text;
- Assignment completed within a selection in the cPP: the completed assignment text is indicated with <u>*italicized and underlined text*</u>
- Refinement: Indicated with **bold** text;
- Selection: Indicated with <u>underlined</u> text;
- Iteration: Indicated by appending the iteration number in parenthesis, e.g., /SigGen.
- Where operations were completed in the NDcPP itself, the formatting used in the NDcPP has been retained.

The following conventions were used to resolve conflicting SFRs between the NDcPP and MOD_VPNGW:

- All SFRs from MOD_VPNGW reproduced as-is
- SFRs that appear in both NDcPP and MOD_VPNGW are modified based on instructions specified in MOD_VPNGW.

## 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 |
|---|---|---|
| FAU: Security audit | FAU_GEN.1 | Audit Data Generation |
| | FAU_GEN.2 | User identity association |
| | FAU_STG_EXT.1 | Protected Audit Event Storage |
| FCS: Cryptographic support | FCS_CKM.1 | Cryptographic Key Generation (Refined) |
| | FCS_CKM.1/IKE | Cryptographic Key Generation (for IKE peer authentication) |
| | FCS_CKM.2 | Cryptographic Key Establishment (Refined) |
| | FCS_CKM.4 | Cryptographic Key Destruction |
| | FCS_COP.1/DataEncryption | Cryptographic Operation (for data encryption/decryption) |
| | FCS_COP.1/SigGen | Cryptographic Operation (for cryptographic signature) |
| | FCS_COP.1/Hash | Cryptographic Operation (for cryptographic hashing) |
| | FCS_COP.1/KeyedHash | Cryptographic Operation (for keyed-hash message authentication) |
| | FCS_IPSEC_EXT.1 | Extended: IPsec |
| | FCS_SSHS_EXT.1 | SSH Server Protocol |
| | FCS_RBG_EXT.1 | Random Bit Generation |
| FIA: Identification and authentication | FIA_AFL.1 | Authentication Failure Handling |
| | FIA_PMG_EXT.1 | Password Management |
| | FIA_UIA_EXT.1 | User Identification and Authentication |

| Class Name | Component Identification | Component Name |
|---|---|---|
| | FIA_UAU_EXT.2 | Password-based Authentication Mechanism |
| | FIA_UAU.7 | Protected Authentication Feedback |
| | FIA_X509_EXT.1/Rev | X.509 Certificate Validation |
| | FIA_X509_EXT.2 | X.509 Certificate Authentication |
| | FIA_X509_EXT.3 | X.509 Certificate Requests |
| | FIA_PSK_EXT.1 | Extended: Pre-Shared Key Composition |
| FMT: Security management | FMT_MOF.1/Services | Trusted Update - Management of TSF Data |
| | FMT_MOF.1/ManualUpdate | Trusted Update - Management of security functions behaviour |
| | FMT_MTD.1/CryptoKeys | Management of TSF Data |
| | FMT_MTD.1/CoreData | Management of TSF Data |
| | FMT_MOF.1/Functions | Management of Security Functions Behaviour |
| | FMT_SMF.1 | Specification of Management Functions |
| | FMT_SMF.1/VPN | Specification of Management Functions (VPN Gateway) |
| | FMT_SMR.2 | Restrictions on security roles |
| FPF: Packet Filtering | FPF_RUL_EXT.1 | Packet Filtering |
| FPT: Protection of the TSF | FPT_SKP_EXT.1 | Protection of TSF Data (for reading of all symmetric keys) |
| | FPT_APW_EXT.1 | Protection of Administrator Passwords |
| | FPT_TST_EXT.1 | Extended: TSF Testing |
| | FPT_TST_EXT.3 | TSF Self-Test with Defined Methods |
| | FPT_TUD_EXT.1 | Extended: Trusted Update |
| | FPT_STM_EXT.1 | Reliable Time Stamps |
| | FPT_FLS.1/SelfTest | Fail Secure |
| 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 | Inter-TSF trusted channel |
| | FTP_ITC.1/VPN | Inter-TSF Trusted Channel (VPN Communications) |
| | FTP_TRP.1/Admin | Trusted Path |

## 5.3 SFRs from NDcPP and MOD_VPNGW

### 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 shut-down of the audit functions;
b) All auditable events for the not specified level of audit; and
*c) All administrator actions comprising:*
   - *Administrative login and logout (name of user account shall be logged if individual user accounts are required for administrators).*
   - *Changes to TSF data related to configuration changes (in addition to the information that a change occurred it shall be logged what has been changed).*
   - *Generating/import of, changing, or deleting of cryptographic keys (in addition to the action itself a unique key name or key reference shall be logged).*
   - *Resetting passwords (name of related user account shall be logged).*

- *[Starting and stopping services [Security related configuration changes (in addition to the information that a change occurred it shall be logged what has been changed)]];*

d) *Specifically defined auditable events listed in Table 16*.

**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

b) For each audit event type, based on the auditable event definitions of the functional components included in the cPP/ST, *information specified in column three of Table 16.*

**Table 16 Auditable Events**

| SFR | Auditable Event | Additional Audit Record Contents |
|---|---|---|
| FAU_GEN.1 | None. | None. |
| FAU_GEN.2 | None. | None. |
| FAU_STG_EXT.1 | None. | None. |
| FCS_CKM.1 | None. | None. |
| FCS_CKM.1/IKE | None. | None. |
| FCS_CKM.2 | None. | None. |
| FCS_CKM.4 | None. | None. |
| FCS_COP.1/DataEncryption | None. | None. |
| FCS_COP.1/SigGen | None. | None. |
| FCS_COP.1/Hash | None. | None. |
| FCS_COP.1/KeyedHash | None. | None. |
| FCS_IPSEC_EXT.1 | Failure to establish an IPsec SA.<br><br>Session establishment with peer | Reason for failure.<br><br>Entire packet contents of packets transmitted/received during session establishment |
| FCS_RBG_EXT.1 | None. | None. |
| FCS_SSHS_EXT.1 | Failure to establish an SSH session | Reason for failure. |
| FIA_AFL.1 | Unsuccessful login attempts limit is met or exceeded. | Origin of the attempt (e.g., IP address) |
| FIA_PMG_EXT.1 | None. | None. |
| FIA_UIA_EXT.1 | All use of the identification and authentication mechanism. | Origin of the attempt (e.g., IP address) |
| FIA_UAU_EXT.2 | All use of the identification and authentication mechanism. | Origin of the attempt (e.g., IP address). |
| FIA_UAU.7 | None. | None. |
| FIA_X509_EXT.1/Rev | Unsuccessful attempt to validate a certificate<br>Any addition, replacement, or removal of trust anchors in the TOE's trust store<br><br>Session Establishment with CA | Reason for failure of certificate validation<br>Identification of certificates added, replaced, or removed as trust anchor in the TOE's trust store<br><br>Entire packet contents of packets transmitted/received during session establishment |
| FIA_X509_EXT.2 | None. | None. |
| FIA_X509_EXT.3 | None. | None. |
| FMT_MOF.1/ManualUpdate | Any attempt to initiate a manual update | None. |
| FMT_MOF.1/Services | Starting and stopping of Services | None. |

| SFR | Auditable Event | Additional Audit Record Contents |
|---|---|---|
| FMT_MTD.1/CoreData | None. | None. |
| FMT_MTD.1/CryptoKeys | None. | None. |
| FMT_SMF.1 | All management activities of TSF data | None. |
| FMT_SMR.2 | None. | None. |
| FPF_RUL_EXT.1 | Application of rules configured with the 'log' operation | Source and destination addresses<br>Source and destination ports<br>Transport Layer Protocol |
| FPT_SKP_EXT.1 | None. | None. |
| FPT_APW_EXT.1 | None. | None. |
| FPT_STM_EXT.1 | Discontinuous changes to time - either Administrator actuated or changed via an automated process. (Note that no continuous changes to time need to be logged. See also application note on FPT_STM_EXT.1) | For discontinuous changes to time: The old and new values for the time.<br>Origin of the attempt to change time for success and failure (e.g., IP address). |
| FPT_TUD_EXT.1 | Initiation of update. result of the update attempt (success or failure) | None. |
| FPT_TST_EXT.1 | None. | None. |
| FPT_TST_EXT.3 | Indication that TSF self-test was completed | None. |
| | Failure of self-test | Reason for failure (including identifier of invalid certificate) |
| FTA_SSL_EXT.1 | The termination of a local session by the session locking mechanism. | None. |
| FTA_SSL.3 | The termination of a remote session by the session locking mechanism. | None. |
| FTA_SSL.4 | The termination of an interactive session. | None. |
| FTA_TAB.1 | None. | None. |
| FTP_ITC.1 | Initiation of the trusted channel.<br>Termination of the trusted channel.<br>Failure of the trusted channel functions. | Identification of the initiator and target of failed trusted channels establishment attempt |
| FTP_TRP.1/Admin | Initiation of the trusted path. Termination of the trusted path. Failure of the trusted path functions. | None. |

### 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 Protected Audit Event 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 according to FTP_ITC.1.

**FAU_STG_EXT.1.2** The TSF shall be able to store generated audit data on the TOE itself. In addition [

- TOE shall consist of a single standalone component that stores audit data locally].

**FAU_STG_EXT.1.3** The TSF shall [overwrite previous audit records according to the following rule: [*the newest audit record will overwrite the oldest audit record.*]] when the local storage space for audit data is full.

## 5.3.2 Cryptographic Support (FCS)

### 5.3.2.1 FCS_CKM.1 Cryptographic Key Generation (Refined)

**FCS_CKM.1.1** The TSF shall generate **asymmetric** cryptographic keys in accordance with a specified cryptographic key generation algorithm: [

- RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3;
- ECC schemes using "NIST curves" [P-256, P-384] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4;
- FFC Schemes using 'safe prime' groups that meet the following: "NIST Special Publication 800-56A Revision 3, Recommendation for Pair-Wise Key establishment Schemes Using Discrete Logarithm Cryptography" and [selection: RFC 3526]

*] ~~and specified cryptographic key sizes [assignment: cryptographic key sizes] that meet the following: [assignment: list of standards]~~.*

### 5.3.2.2 FCS_CKM.1/IKE Cryptographic Key Generation (for IKE peer authentication)

**FCS_CKM.1.1/IKE** The TSF shall generate **asymmetric** cryptographic keys **used for IKE peer authentication** in accordance with a specific cryptographic key generation algorithm:

[

- **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 [no other curves]**]

**and [**

- **FFC Schemes using "safe-prime" groups that meet the following: 'NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" and [selection: RFC 3526]]**

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

### 5.3.2.3 FCS_CKM.2 Cryptographic Key Establishment (Refined)

**FCS_CKM.2.1** The TSF shall **perform** cryptographic **key establishment** in accordance with a specified cryptographic key **establishment** method: [

- Elliptic curve-based key establishment schemes that meets the following: NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography";

- FFC Schemes using “safe-prime” groups that meet the following: NIST Special Publication 800-56A Revision 3, “Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography” and [selection: RFC 3526]

*]* ~~*that meets the following: [assignment: list of standards]*~~.

### 5.3.2.4 FCS_CKM.4 Cryptographic Key Destruction

**FCS_CKM.4.1** The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method

- For plaintext keys in volatile storage, the destruction shall be executed by a [single overwrite consisting of *[zeroes]*];
- For plaintext keys in non-volatile storage, the destruction shall be executed by the invocation of an interface provided by a part of the TSF that [
    - logically addresses the storage location of the key and performs a [single] overwrite consisting of *[zeroes]];*

that meets the following: *No Standard*.

### 5.3.2.5 FCS_COP.1/DataEncryption Cryptographic Operation (AES Data Encryption/Decryption)

**FCS_COP.1.1/DataEncryption** The TSF shall perform *encryption/decryption* in accordance with a specified cryptographic algorithm *AES used in* [*CBC, GCM*] *mode* and cryptographic key sizes [*128 bits, 192 bits, 256 bits*] that meet the following: *AES as specified in ISO 18033-3,* [CBC as specified in ISO 10116, GCM as specified in ISO 19772].

### 5.3.2.6 FCS_COP.1/SigGen Cryptographic Operation (Signature Generation and Verification)

**FCS_COP.1.1/SigGen** The TSF shall perform *cryptographic signature services (generation and verification)* in accordance with a specified cryptographic algorithm

[

- RSA Digital Signature Algorithm and cryptographic key sizes (modulus) [*2048 bits or greater*],

- Elliptic Curve Digital Signature Algorithm and cryptographic key sizes [*256 bits or greater*]

that meet the following: *[*

- For RSA schemes: FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSAPKCS1v1_5; ISO 9796-2, Digital signature scheme 2 or Digital Signature scheme 3,

- For ECDSA schemes: FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Section 6 and Appendix D, Implementing “NIST curves” [P-256, P-384]; ISO 14888-3, Section 6.4

*]*.

### 5.3.2.7 FCS_COP.1/Hash Cryptographic Operation (Hash Algorithm)

**FCS_COP.1.1/Hash** The TSF shall perform *cryptographic hashing services* in accordance with a specified cryptographic algorithm [SHA-1, SHA-256, SHA-384, SHA-512] ~~and cryptographic key sizes [assignment: cryptographic key sizes]~~ and **message digest sizes [160, 256, 384, 512] bits** that meet the following: *ISO 10118-3:2004*.

### 5.3.2.8 FCS_COP.1/KeyedHash Cryptographic Operation (Keyed Hash Algorithm)

**FCS_COP.1.1/KeyedHash** The TSF shall perform *keyed-hash message authentication* in accordance with a specified cryptographic algorithm [HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-512] and cryptographic key sizes [*160-bit, 256-bit, 512-bit*] **and message digest sizes [160, 256, 512] bits** that meet the following: *ISO 9797-2:2011, Section 7 "MAC Algorithm 2".*

### 5.3.2.9 FCS_IPSEC_EXT.1 Extended: IPSEC

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

**FCS_IPSEC_EXT.1.2** The TSF shall have a nominal, final entry in the SPD that matches anything that is otherwise unmatched and discards it.

**FCS_IPSEC_EXT.1.3** The TSF shall implement [transport mode, tunnel mode].

**FCS_IPSEC_EXT.1.4** The TSF shall implement the IPsec protocol ESP as defined by RFC 4303 using the cryptographic algorithms [AES-CBC-128 (RFC 3602), AES-CBC-256 (RFC 3602), AES-GCM-128 (RFC 4106), AES-GCM-256 (RFC 4106)] and [AES-CBC-192 (RFC3602), AES-GCM-192 (RFC 4106)] together with a Secure Hash Algorithm (SHA)-based HMAC [HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-512].

**FCS_IPSEC_EXT.1.5** The TSF shall implement the protocol: [

- IKEv2 as defined in RFC 5996 and [with mandatory support for NAT traversal as specified in RFC 5996, section 2.23)], and [RFC 4868 for hash functions]

].

**FCS_IPSEC_EXT.1.6** The TSF shall ensure the encrypted payload in the [IKEv2] protocol uses the cryptographic algorithms [AES-CBC-128, AES-CBC-192, AES-CBC-256 (specified in RFC 3602), AES-GCM-128, AES-GCM-256 (specified in RFC 5282)].

**FCS_IPSEC_EXT.1.7** The TSF shall ensure that [

- IKEv2 SA lifetimes can be configured by a Security Administrator based on

  [

    - length of time, where the time values can be configured within [*1-24*] hours;

  ].

]

**FCS_IPSEC_EXT.1.8** The TSF shall ensure that [

- IKEv2 Child SA lifetimes can be configured by a Security Administrator based on

  [

    - number of bytes
    - length of time, where the time values can be configured within [*1-8*] hours

  ]

].

**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 [*224 (for DH Group 14), 256 (for DH Group 19), and 384 (for DH Group 20)*] *bits*.

**FCS_IPSEC_EXT.1.10** The TSF shall generate nonces used in [IKEv2] exchanges of length [

- according to the security strength associated with the negotiated Diffie-Hellman group;
- at least 128 bits in size and at least half the output size of the negotiated pseudorandom function (PRF) hash
  ].

**FCS_IPSEC_EXT.1.11** The TSF shall ensure that IKE protocols implement DH Group(s)
[

- [14 (2048-bit MODP)] according to RFC 3526,
- [19 (256-bit Random ECP),
- 20 (384-bit Random ECP] according to RFC 5114.
  ].

**FCS_IPSEC_EXT.1.12** 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 [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 [IKEv2 CHILD_SA] connection.

**FCS_IPSEC_EXT.1.13** The TSF shall ensure that all IKE protocols perform peer authentication using [RSA, ECDSA] that use X.509v3 certificates that conform to RFC 4945 and [Pre-shared Keys].

**FCS_IPSEC_EXT.1.14** The TSF shall only establish a trusted channel if the presented identifier in the received certificate matches the configured reference identifier, where the presented and reference identifiers are of the following fields and types: [**Distinguished Name (DN),** CN: IP address, CN: Fully Qualified Domain Name (FQDN), CN: user FQDN, **no other reference identifier type**].

### 5.3.2.10 FCS_RBG_EXT.1 Random Bit Generation

**FCS_RBG_EXT.1.1** The TSF shall perform all deterministic random bit generation services in accordance with ISO 18031:2011 using [CTR_DRBG (AES)].

**FCS_RBG_EXT.1.2** The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [[*1*] platform-based noise source] with a minimum of [256 bits] of entropy at least equal to the greatest security strength, according to ISO 18031:2011 Table C.1 “Security Strength Table for Hash Functions”, of the keys and hashes that it will generate.

### 5.3.2.11 FCS_SSHS_EXT.1 SSH Server Protocol

**FCS_SSHS_EXT.1.1** The TSF shall implement the SSH protocol in accordance with: RFCs 4251, 4252, 4253, 4254 [ 6668*]*.

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

**FCS_SSHS_EXT.1.3** The TSF shall ensure that, as described in RFC 4253, packets greater than [*32,765*] bytes in an SSH transport connection are dropped.

**FCS_SSHS_EXT.1.4** The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms and rejects all other encryption algorithms: [aes128-cbc, aes256-cbc].

**FCS_SSHS_EXT.1.5** The TSF shall ensure that the SSH public-key based authentication implementation uses [ssh-rsa] as its public key algorithm(s) and rejects all other public key algorithms.

**FCS_SSHS_EXT.1.6** The TSF shall ensure that the SSH transport implementation uses [hmac-sha1, hmac-sha2-256, hmac-sha2-512] as its MAC algorithm(s) and rejects all other MAC algorithm(s).

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

**FCS_SSHS_EXT.1.8** The TSF shall ensure that within SSH connections, the same session keys are used for a threshold of no longer than one hour, and each encryption key is used to protect no more than one gigabyte of data. After any of the thresholds are reached, a rekey needs to be performed.

## 5.3.3 Identification and authentication (FIA)

### 5.3.3.1 FIA_AFL.1 Authentication Failure Management

**FIA_AFL.1.1** The TSF shall detect when an Administrator configurable positive integer within [*1 to 25*] unsuccessful authentication attempts occur related to *Administrators attempting to authenticate remotely using a password*.

**FIA_AFL.1.2:** When the defined number of unsuccessful authentication attempts has been met, the TSF shall [prevent the offending administrator from successfully establishing a remote session using any authentication method that involves a password until [*an authorized administrator unlocks the locked user account*] is taken by an Administrator].

### 5.3.3.2 FIA_PMG_EXT.1 Password Management

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

a) Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters*:* [“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“,”)”,];
b) Minimum password length shall be configurable to between [*15*] and [*15*] characters.

### 5.3.3.3 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.

### 5.3.3.4 FIA_UAU_EXT.2 Password-based Authentication Mechanism

**FIA_UAU_EXT.2.1** The TSF shall provide a local [password-based] authentication mechanism to perform local administrative user authentication.

### 5.3.3.5 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**.

### 5.3.3.6 FIA_X509_EXT.1/Rev X.509 Certificate Validation

**FIA_X509_EXT.1.1/Rev** The TSF shall validate certificates in accordance with the following rules:

- RFC 5280 certificate validation and certificate path validation **supporting a minimum path length of three certificates.**
- The certificate path must terminate with a trusted CA certificate designed as a trust anchor.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of the certificate using [a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3].
- The TSF shall validate the extendedKeyUsage field according to the following rules:
  - *Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.*
  - *Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.*
  - *Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.*
  - *OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.*

**FIA_X509_EXT.1.2/Rev** The TSF shall only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.

### 5.3.3.7 FIA_X509_EXT.2 X.509 Certificate Authentication

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

**FIA_X509_EXT.2.2** When the TSF cannot establish a connection to determine the validity of a certificate, the TSF shall [not accept the certificate].

### 5.3.3.8 FIA_X509_EXT.3 X.509 Certificate Requests

**FIA_X509_EXT.3.1** The TSF shall generate a Certificate Request as specified by RFC 2986 and be able to provide the following information in the request: public key and [Common Name, Organization, Organizational Unit, Country].

**FIA_X509_EXT.3.2** The TSF shall validate the chain of certificates from the Root CA upon receiving the CA Certificate Response.

#### 5.3.3.9 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 between 22 and 127 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*].

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

### 5.3.4 Security management (FMT)

#### 5.3.4.1 FMT_MOF.1/ManualUpdate Management of security functions behavior

**FMT_MOF.1/ManualUpdate** The TSF shall restrict the ability to enable the functions *to perform manual update to Security Administrators*.

#### 5.3.4.2 FMT_MOF.1/ Functions Management of Security Functions Behaviour

**FMT_MOF.1.1/Functions** The TSF shall restrict the ability to [modify the behaviour of] the functions [transmission of audit data to an external IT entity] to *Security* Administrators.

#### 5.3.4.3 FMT_MOF.1/ Services Management of Security Functions Behavior

**FMT_MOF.1.1/Services** The TSF shall restrict the ability to **start and stop** ~~the functions~~ **services** to *Security Administrators*.

#### 5.3.4.4 FMT_MTD.1/CoreData Management of TSF Data

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

#### 5.3.4.5 FMT_MTD.1/CryptoKeys Management of TSF data

**FMT_MTD.1.1/CryptoKeys** The TSF shall restrict the ability to [[*manage*]] the [*cryptographic keys* ***and certificates used for VPN operation]*** to *[Security Administrators].*

#### 5.3.4.6 FMT_SMF.1 Specification of Management Functions

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

- Ability to administer the TOE locally and remotely;
- Ability to configure the access banner;
- Ability to configure the session inactivity time before session termination or locking;
- Ability to update the TOE, and to verify the updates using **digital signature and [hash comparison]** capability prior to installing those updates;

- Ability to configure the authentication failure parameters for FIA_AFL.1;
- **Ability to manage the cryptographic keys;**
- **Ability to configure the cryptographic functionality;**
- **Ability to configure the lifetime for the IPsec SAs;**
- **Ability to import X.509v3 certificates to the TOE's trust store;**

[

- Ability to start and stop services;
- Ability to configure audit behaviour (e.g., changes to storage locations for audit; changes to behaviour when local audit storage space is full);
- Ability to modify the behaviours of the transmission of audit data to an external IT entity, the handling of audit data, the audit functionality when Local Audit Storage Space is full;
- Ability to configure the list of TOE-provided services available before an entity is identified and authenticated as specified in FIA_UIA_EXT.1;
- Ability to configure thresholds for SSH rekeying;Ability to enable or disable automatic checking for updates or automatic updates;
- Ability to re-enable an Administrator account;
- Ability to set the time which is used for time-stamps;
- Ability to configure the reference identifier for the peer;
- Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors;
- Ability to manage the trusted public keys database;
- No other capabilities]].

#### 5.3.4.7 FMT_SMF.1/VPN Specification of Management Functions (VPN Gateway)

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

- Definition of packet filtering rules;
- Association of packet filtering rules to network interfaces;
- Ordering of packet filtering rules by priority;

[

- [no other capabilities]].

#### 5.3.4.8 FMT_SMR.2 Restrictions on Security Roles

**FMT_SMR.2.1** The TSF shall maintain the roles:

- *Security 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

- *The Security Administrator role shall be able to administer the TOE locally;*
- *The Security Administrator role shall be able to administer the TOE remotely*

are satisfied.

## 5.3.5 Packet Filtering (FPF)

### 5.3.5.1 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 allow the definition of Packet Filtering rules using the following network protocols and protocol fields:

- IPv4 (RFC 791)
  - Source address
  - Destination Address
  - Protocol
- IPv6 (RFC 2460)
  - Source address
  - Destination Address
  - Next Header (Protocol)
- TCP (RFC 793)
  - Source Port
  - Destination Port
- UDP (RFC 768)
  - Source Port
  - Destination Port

**FPF_RUL_EXT.1.3** The TSF shall allow the following operations to be associated with Packet Filtering rules: permit and drop with the capability to log the operation.

**FPF_RUL_EXT.1.4** The TSF shall allow the Packet Filtering rules to be assigned to each distinct network interface.

**FPF_RUL_EXT.1.5** The TSF shall process the applicable Packet Filtering rules (as determined in accordance with FPF_RUL_EXT.1.4) in the following order: Administrator-defined.

**FPF_RUL_EXT.1.6** The TSF shall drop traffic if a matching rule is not identified.

## 5.3.6 Protection of the TSF (FPT)

### 5.3.6.1 FPT_FLS.1/SelfTest Fail Secure (Self-Test Failures)

**FPT_FLS.1.1/SelfTest** The TSF shall **shut down** 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.*]

### 5.3.6.2 FPT_SKP_EXT.1: Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)

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

### 5.3.6.3 FPT_APW_EXT.1 Extended: Protection of Administrator Passwords

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

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

### 5.3.6.4 FPT_STM_EXT.1 Reliable Time Stamps

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

**FPT_STM_EXT.1.2** The TSF shall [allow the Security Administrator to set the time].

### 5.3.6.5 FPT_TST_EXT.1: TSF Testing

**FPT_TST_EXT.1.1** The TSF shall run a suite of the following self-tests [during initial start-up (on power on). periodically during normal operation] to demonstrate the correct operation of the TSF: **noise source health tests,** [

- *AES Known Answer Test*
- *RSA Signature Known Answer Test (both signature/verification)*
- *RNG/DRBG Known Answer Test*
- *HMAC Known Answer Test*
- *SHA-1/256/384/512 Known Answer Test*
- *ECDSA self-test*
- *Software Integrity Test*

].

### 5.3.6.6 FPT_TST_EXT.3: TSF Self-Test with Defined Methods

**FPT_TST_EXT.3.1** The TSF shall run a suite of the following self-tests [*[when loaded for execution]*] to demonstrate the correct operation of the TSF: [*integrity verification of stored executable code*].

**FPT_TST_EXT.3.2** The TSF shall execute the self-testing through [*a TSF-provided cryptographic service specified in FCS_COP.1/SigGen*].

### 5.3.6.7 FPT_TUD_EXT.1 Extended: Trusted Update

**FPT_TUD_EXT.1.1** The TSF shall provide *Security Administrators* the ability to query the currently executing version of the TOE firmware/software and [no other TOE firmware/software version].

**FPT_TUD_EXT.1.2** The TSF shall provide *Security Administrators* the ability to manually initiate updates to TOE firmware/software and [no other update mechanism].

**FPT_TUD_EXT.1.3** The TSF shall provide means to authenticate firmware/software updates to the TOE using a **digital signature mechanism and *[*<u>*published hash*</u>*]*** prior to installing those updates.

## 5.3.7 TOE Access (FTA)

### 5.3.7.1 FTA_SSL_EXT.1 TSF-initiated Session Locking

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

- <u>terminate the session</u>]

after a Security Administrator-specified time period of inactivity.

### 5.3.7.2 FTA_SSL.3 TSF-initiated Termination

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

### 5.3.7.3 FTA_SSL.4 User-initiated Termination

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

### 5.3.7.4 FTA_TAB.1 Default TOE Access Banners

**FTA_TAB.1.1:** 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 (Refinement)

**FTP_ITC.1.1** The TSF **shall be capable of using** [**<u>IPsec</u>] to** provide a trusted communication channel between itself and **authorized IT entities supporting the following capabilities: audit server [<u>authentication server</u>, *[<u>CA Server</u>]*]** 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 IPsec,*
- *a CA server using IPsec].*

### 5.3.8.2 FTP_ITC.1/VPN Inter-TSF Trusted Channel (VPN Communications)

**FTP_ITC.1.1/VPN** The TSF shall **be capable of using IPsec to** provide a communication channel between itself and **authorized IT entities supporting VPN communications** 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.1.2VPN** The TSF shall permit **[***the authorized IT entities*] to initiate communication via the trusted channel.

**FTP_ ITC.1.3/VPN** The TSF shall initiate communication via the trusted channel for [remote VPN gateways/peers].

### 5.3.8.3 FTP_TRP.1/Admin Trusted Path

**FTP_TRP.1.1/Admin:** The TSF shall **be capable of using [SSH] to** provide a communication path between itself and **authorized** 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 provides detection of modification of the channel data**.

**FTP_TRP.1.2/Admin** The TSF shall permit remote **Administrators** to initiate communication via the trusted path.

**FTP_TRP.1.3/Admin** 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 PP

The NDcPP v2.2e and MOD_VPNGW v1.1 contain all the requirements claimed in this Security Target. As such the dependencies are not applicable since the PP and EP have been approved.

## 5.5 Security Assurance Requirements

### 5.5.1 SAR Requirements

The TOE assurance requirements for this ST are taken directly from the NDcPP which are derived from Common Criteria Version 3.1, Revision 5. The assurance requirements are summarized in the table below:

**Table 17 Assurance Measures**

| Assurance Class | Components | Components Description |
|---|---|---|
| Security Target (ASE) | ASE_CCL.1 | Conformance claims |
| | ASE_ECD.1 | Extended components definition |
| | ASE_INT.1 | ST introduction |
| | ASE_OBJ.1 | Security objectives for the operational environment |
| | ASE_REQ.1 | Stated security requirements |
| | ASE_SPD.1 | Security Problem Definition |
| | ASE_TSS.1 | TOE summary specification |
| Development (ADV) | ADV_FSP.1 | Basic Functional Specification |
| Guidance documents (AGD) | AGD_OPE.1 | Operational user guidance |
| | AGD_PRE.1 | Preparative User guidance |
| Life cycle support (ALC) | ALC_CMC.1 | Labeling of the TOE |
| | ALC_CMS.1 | TOE CM coverage |
| Tests (ATE) | ATE_IND.1 | Independent testing - sample |
| Vulnerability assessment (AVA) | AVA_VAN.1 | Cisco will provide a list of TOE hardware and software components. The lab will conduct the additional vulnerability testing as prescribed by the MOD_VPNGW_v1.0. |

### 5.5.2 Security Assurance Requirements Rationale

The Security Assurance Requirements (SARs) in this Security Target represent the SARs identified in the NDcPP v2.2e and MOD_VPNGW v1.1. As such, the NDcPP SAR rationale is deemed acceptable since the PPs have been validated.

## 5.6 Assurance Measures

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

**Table 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 behaviour 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<br>ALC_CMS.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. The TOE will also be provided along with the appropriate administrative guidance. |
| 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

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

**Table 19 How TOE SFRs Measures**

| TOE SFRs | How the SFR is Met |
|---|---|
| FAU_GEN.1 | The TOE generates an audit record whenever an audited event occurs. The types of events that cause audit records to be generated include: startup and shutdown of the audit mechanism cryptography related events to include IPsec session establishment with peer, identification and authentication related events, and administrative events (the specific events and the contents of each audit record are listed in the table within the FAU_GEN.1 SFR, "Auditable Events Table"). Each of the events is specified in syslog records in enough detail to identify the user for which the event is associated, when the event occurred, where the event occurred, the outcome of the event, and the type of event that occurred such as generating keys, including the type of key. Additionally, the startup and shutdown of the audit functionality is audited.<br><br>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:<br><br>Nov 19 13:55:59: %CRYPTO-6-SELF_TEST_RESULT: Self test info: (Self test activated by user: lab)<br>Nov 19 13:55:59: %CRYPTO-6-SELF_TEST_RESULT: Self test info: (Software checksum ... passed)<br>Nov 19 13:55:59: %CRYPTO-6-SELF_TEST_RESULT: Self test info: (DES encryption/decryption ... passed)<br>Nov 19 13:55:59: %CRYPTO-6-SELF_TEST_RESULT: Self test info: (3DES encryption/decryption ... passed)<br>Nov 19 13:55:59: %CRYPTO-6-SELF_TEST_RESULT: Self test info: (SHA hashing ... passed)<br>Nov 19 13:55:59: %CRYPTO-6-SELF_TEST_RESULT: Self test info: (AES encryption/decryption ... passed)<br><br>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"<br><br>The logging buffer size can be configured from a range of 4096 (default) to 4,294,967,295 bytes. It is noted to not make the buffer size too large because the TOE could run out of memory for other tasks. Use the show memory privileged EXEC command to view the free processor memory on the TOE. However, this value is the maximum available, and the buffer size should not be set to this amount.<br><br>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).<br><br>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. The logs can be saved to flash memory, so records are not lost in case of failures or restarts. Refer to |

| TOE SFRs | How the SFR is Met |
|---|---|
| | the Common Criteria Operational User Guidance and Preparative Procedures for command description and usage information.<br><br>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 messages can be sent to the console alerting the administrator that some action needs to be taken as these types of messages mean that the functionality of the TOE is affected. All notifications and information type message can be sent to the syslog server. The audit records are transmitted using IPsec tunnel to the syslog server. If the communications to the syslog server is lost, the TOE generates an audit record, and all permit traffic is denied until the communications is re-established.<br><br>The FIPS crypto tests performed during startup, the messages are displayed only on the console. Once the box is up and operational and the crypto self-test command is entered, then the messages are displayed on the console and will also be logged. For the TSF self-test, successful completion of the self-test is indicated by reaching the log-on prompt. If there are issues, the applicable audit record is generated and displayed on the console.<br><br>When the incoming traffic to the TOE exceeds what the interface can handle, the packets are dropped at the input queue itself, and for each interface, the TOE indicates the number of dropped packets. |
| FAU_GEN.2 | 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:<br><br>Jun 18 11:17:20.769: AAA/BIND(0000004B): Bind i/f<br>Jun 18 11:17:20.769: AAA/AUTHEN/LOGIN (0000004B): Pick method list 'default'<br>Jun 18 2012 11:17:26 UTC: %SEC_LOGIN-5-LOGIN_SUCCESS: Login Success [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 in real-time. The TOE protects communications with an external syslog server via IPsec. If the IPsec connection fails, the TOE will store audit records on the TOE when it discovers it can no longer communicate with its configured syslog server. When the connection is restored, the TOE will transmit the buffer contents when connected to the syslog server.<br><br>For audit records stored internally to the TOE the audit records are stored in a circular log file where the TOE overwrites the oldest audit records when the audit trail becomes full. The size of the logging files on the TOE is configurable by the administrator with the minimum value being 4096 (default) to 2147483647 bytes of available disk space Refer to the Common Criteria Operational User Guidance and Preparative Procedures for command description and usage information.<br><br>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. |
| FCS_CKM.1<br>FCS_CKM.1/IKE<br>FCS_CKM.2 | The TOE implements DH group 14 key establishment schemes that meets RFC 3526, Section 3, Elliptic curve key establishment (conformant to NIST SP 800-56A). The TOE acts as both a sender and receiver for Diffie-Helman based key establishment schemes.<br><br>The TOE complies with section 5.6 and all subsections regarding asymmetric key pair generation and key establishment in the NIST SP 800-56A and with section 6. |

<table>
<tr><th>TOE SFRs</th><th>How the SFR is Met</th></tr>
<tr><td></td><td>Asymmetric cryptographic keys used for IKE peer authentication are generated according to FIPS PUB 186-4, Appendix B.3 for RSA schemes and Appendix B.4 for ECDSA schemes.<br><br>The TOE can create an RSA public-private key pair, with a minimum RSA key size of 2048-bit (for CSfC purposes, the TOE is capable of a minimum RSA key size of 3072-bit) and ECDSA key pairs using NIST curves P-256 and P-384. Both RSA and ECC schemes can be used to generate a Certificate Signing Request (CSR).<br><br>
<table>
<tr><th>Scheme</th><th>SFR</th><th>Service</th></tr>
<tr><td rowspan="2">RSA<br>Key generation</td><td>FCS_SSHS_EXT.1</td><td>SSH Remote Administration</td></tr>
<tr><td>FCS_IPSEC_EXT.1</td><td>Transmit generated audit data to an external IT entity</td></tr>
<tr><td>ECC<br>Key generation<br>Key establishment</td><td>FCS_IPSEC_EXT.1</td><td>Transmit generated audit data to an external IT entity</td></tr>
<tr><td rowspan="2">FFC<br>Key generation<br>Key establishment</td><td>FCS_SSHS_EXT.1</td><td>SSH Remote Administration</td></tr>
<tr><td>FCS_IPSEC_EXT.1</td><td>Transmit generated audit data to an external IT entity</td></tr>
</table>
<br>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 X.509v3 certificate from the CA. Integrity of the CSR and certificate during transit are assured through use of digital 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.<br><br>The TOE can also use the X.509v3 certificate for securing IPsec sessions. The TOE provides cryptographic signature services using ECDSA that meets FIPS 186-4, "Digital Signature Standard" with NIST curves P-256, P-384, and RSA that meets FIPS PUB 186-4, "Digital Signature Standard".</td></tr>
<tr><td>FCS_CKM.4</td><td>The TOE destroys all keys and Critical Security Parameters (CSPs) when no longer required for use. See Table 21: TOE Key Zeroization in Section 7 Key Zeroization. The information provided in the table includes all of the secrets, keys and associated values, the description, and the method used to zeroize keys/CSPs when no longer required for use.<br><br>The information is provided in the reference section for ease and readability of all of the secrets, keys and associated values, their description and zeroization methods.</td></tr>
<tr><td>FCS_COP.1/DataEncryption</td><td>The TOE provides symmetric encryption and decryption capabilities using AES in GCM and CBC mode (128, 192 and 256 bits) as described in ISO 18033-3, ISO 19772, and ISO 10116 respectively. Please see CAVP certificate in Table 6 above for validation details. AES is implemented in the following protocols: IPsec and SSH.</td></tr>
<tr><td>FCS_COP.1/SigGen</td><td>The TOE provides cryptographic signature services using RSA Digital Signature Algorithm with key size of 2048 and greater as specified in ISO 9796-2, Digital signature scheme 2 or Digital Signature scheme 3. In addition, the TOE will provide cryptographic signature services using ECDSA with key size of 256 or greater as specified in FIPS PUB 186-4, "Digital Signature Standard". The TOE provides cryptographic signature services using ECDSA that meets ISO 14888-3, Section 6.4 with NIST curves P-256 and P-384.</td></tr>
</table>

| TOE SFRs | How the SFR is Met |
|---|---|
| FCS_COP.1/Hash<br>FCS_COP.1/KeyedHash | The TOE provides cryptographic hashing services using SHA-1, SHA-256, SHA-384 and SHA-512 as specified in ISO 10118-3:2004.<br><br>The TOE provides keyed-hashing message authentication services using HMAC-SHA-1 and HMAC-SHA-256 operates on 512-bit blocks and HMAC-SHA-512 operate on 1024-bit blocks of data, with key sizes and message digest sizes of 160-bits, 256 bits and 512 bits respectively) as specified in ISO 9797-2:2011, Section 7 "MAC Algorithm 2".<br><br>For IKE (ISAKMP) hashing, administrators can select any of HMAC-SHA-1, HMAC-SHA-256 and/or HMAC-SHA-512 (with message digest sizes of 160, 256 and 512 bits respectively) to be used with remote IPsec endpoints.<br><br>SHA-512 hashing is used for verification of software image integrity.<br><br>The TOE provides Secure Hash Standard (SHS) hashing in support of SSH for secure communications. Management of the cryptographic algorithms is provided through the CLI with auditing of those commands.<br><br>The TOE uses HMAC-SHA1 message authentication as part of the RADIUS Key Wrap functionality.<br><br>For IPsec SA authentication integrity options administrators can select any of esp-sha-hmac (HMAC-SHA-1), esp-sha256-hmac, or esp-sha512-hmac (with message digest sizes of 160 and 256 and 512 bits respectively) to be part of the IPsec SA transform-set to be used with remote IPsec endpoints.<br><br>Please see CAVP certificate in Table 6 for validation details. |
| FCS_RBG_EXT.1 | The TOE implements a NIST-approved AES-CTR Deterministic Random Bit Generator (DRBG), as specified in SP 800-90 seeded by an entropy source that accumulates entropy from a TSF-platform based noise source.<br><br>The deterministic RBG is seeded with a minimum of 256 bits of entropy, which is at least equal to the greatest security strength of the keys and hashes that it will generate. |
| FCS_IPSEC_EXT.1 | The TOE implements IPsec to provide authentication and encryption services to prevent unauthorized viewing or modification of data as it travels over the external network.<br><br>The IPsec implementation provides both VPN peer-to-peer TOE 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.<br><br>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.<br><br>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. The IPsec protocol ESP is implemented using the cryptographic algorithms AES-GCM-128, AES-GCM-192, AES-GCM-256, AES-CBC-128, AES-CBC-192 and AES-CBC-256 together with HMAC-SHA-1, HMAC-SHA-256, and HMAC-SHA-512. |

| TOE SFRs | How the SFR is Met |
|---|---|
| | 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.<br><br>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 strength of the symmetric algorithm negotiated to protect the IKEv2 IKE_SA connection is greater than or equal to the strength of the symmetric algorithm negotiated to protect the IKEv2 CHILD_SA connection. The Security Administrator must configure the IKEv2 Transform Sets to ensure that the symmetric algorithm used in the connection is great or equal to the symmetric algorithm use to protect the IKEv2 CHILD_SA connection. The strength of the IKEv2 IKE_SA connection is checked during tunnel negotiation. The IKE protocols implement Peer Authentication using RSA and ECDSA along with X.509v3 certificates, or pre-shared keys.<br><br>When certificates are used for authentication, the distinguished name (DN) is verified to ensure the certificate is valid and is from a valid entity. The DN naming attributes in the certificate is compared with the expected DN naming attributes and deemed valid if the attribute types are the same and the values are the same and as expected. The fully qualified domain name (FQDN) can also be used as verification where the attributes in the certificate are compared with the expected CN: FQDN, CN: user FQDN and CN: IP Address.<br><br>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:<br><br>• The negotiation of mutually acceptable IPsec options between peers (including peer authentication parameters, either signature based or pre-shared key based),<br>• The establishment of additional Security Associations to protect packets flows using Encapsulating Security Payload (ESP), and<br>• The agreement of secure bulk data encryption AES keys for use with ESP.<br><br>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.<br><br>The TOE supports IKEv2 session establishment. The TOE supports configuration lifetimes of both Phase 1 SAs and Phase 2 SAs using the following command, "lifetime". The time values for Phase 1 SAs can be limited up to 24 hours and for Phase 2 SAs up to 8 hours. The Phase 2 SA lifetimes can also be configured by an Administrator based on number of packets.<br><br>The TOE supports Diffie-Hellman Group 14, 19, and 20. To set the DH group, use the 'group [x]' command replacing the group number for each of the supported groups, for example 'group 14'. The nonces used in IKE exchanges are generated using a NIST-approved AES-CTR DRBG and 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]. The nonce is generated according to the security strength associated with the negotiated DH group, is at least 128 bits in size, and is at least half the output size of the negotiated pseudorandom function (PRF) hash. The PRF is 256- 512-bits in length. Table 20 below maps each DH group to its security strength and output length.<br><br>**Table 20 - DH Group Mapping of Security Strength and Output Length**<br><br>DH Group \| Security Strength \| Output Length |

| TOE SFRs | How the SFR is Met |
|---|---|

| | (bits) | (bits) |
|---|---|---|
| 14<br>(2048-bit MODP) | 112 | L = 2048<br>N = 224 |
| 19<br>(ECDH using NIST curve-256) | 128 | L = 3072<br>N = 256 |
| 20<br>(ECDH using NIST curve P-384) | 192 | L = 7680<br>N = 384 |

The secret value ‘x’ used in the IKE Diffie-Hellman key exchange (“x” in $g^x$ mod p) is generated using a NIST-approved AES-CTR DRBG and shall have possible lengths from 224 – 384-bits (112 – 192-bits security strength).

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 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, AES-CBC-192, AES-CBC-256, AES-GCM-128 AES-GCM-256 for encrypting 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), and 20 (384-bit Random ECP) 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, Table 2 in NIST SP 800-57 “Recommendation for Key Management –Part 1: General” and the following corresponding key sizes (in bits) are used: 224 (for DH Group 14), 256 (for DH Group 19), and 384 (for DH Group 20).

IPsec provides secure tunnels between two peers, such as two routers. 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. 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 (SPD)) 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. Separate access lists define blocking and permitting at the interface). For example:

Router# access-list 101 permit ip 192.168.3.0 0.0.0.255 10.3.2.0 0.0.0.255

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 ipsec-isakmp, IPsec is triggered. For example:

Router# crypto map MAP_NAME 10 ipsec-isakmp

| TOE SFRs | How the SFR is Met |
| --- | --- |
| | The match address 101 command means to use access list 101 in order to determine which traffic is relevant. For example:<br><br>Router# (config-crypto-map)#match address 101<br><br>The traffic matching the permit acls would then flow through the IPsec tunnel and be classified as “PROTECTED”.<br><br>Traffic that does not match a permit acl and is also blocked by other non-crypto acls on the interface would be DISCARDED.<br><br>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. 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.<br><br>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.<br><br>The command “fqdn <name” can be configured within a crypto identity and applied to a crypto map in order to perform validation of the peer device during authentication.<br><br>Certificate maps provide the ability for a certificate to be matched with a given set of criteria. You can specify which fields within a certificate should be checked and which values those fields may or may not have. There are six logical tests for comparing the field with the value: equal, not equal, contains, does not contain, less than, and greater than or equal. ISAKMP and ikev2 profiles can bind themselves to certificate maps, and the TOE will determine if they are valid during IKE authentication. |
| FCS_SSHS_EXT.1 | The TOE implementation of SSHv2 complies with RFCs 4251, 4252, 4253, 4254, 6668 and supports the following:<br><br>• Public key algorithm for authentication is RSA Signature Verification (ssh-rsa) which is configured during the TOE installation. Refer to the Cisco 900 Series Integrated Services Routers (ISR) running IOS v15.9 Common Criteria Operational User Guidance And Preparative Procedures for the configuration options and settings.<br>• The TOE supports ssh-rsa host public key algorithms. The host public key algorithms are configured during the TOE installation. Refer to the Cisco 900 Series Integrated Services Routers (ISR) running IOS v15.9 Common Criteria Operational User Guidance And Preparative Procedures for the configuration options and settings.<br>• Local password-based authentication for administrative users accessing the TOE through SSHv2, and optionally supports deferring password-based authentication to a remote AAA server.<br>• Encryption algorithms, AES-CBC-128, AES-CBC-256 to ensure confidentiality of the session.<br>• The TOE’s implementation of SSHv2 supports hashing algorithms hmac-sha1, hmac-sha256, and hmac-sha512 to ensure the integrity of the session.<br>• The TOE’s implementation of SSHv2 can be configured to only allow Diffie-Hellman Group 14 (2048-bit keys) Key Establishment.<br>• Packets greater than *32,765* bytes in an SSH transport connection are dropped. Large packets are detected by the SSH implementation and dropped internal to the SSH process. |

| TOE SFRs | How the SFR is Met |
| --- | --- |
| | • The TOE can also be configured to ensure that SSH re-key of no longer than one hour and no more than one gigabyte of transmitted data for the session key. Note, the TOE will react to first threshold limit reached and perform an ip-ssh rekey.<br>• When an SSH client presents a public key, the TOE establishes a user identity by verifying that the SSH client's presented public key matches one that is stored within an authorized keys file. |
| FIA_AFL.1 | The TOE provides the privileged administrator the ability to specify the maximum number of unsuccessful authentication attempts before privileged administrator or non-privileged administrator is locked out through the administrative CLI using a privileged CLI command. While the TOE supports a range from 1-25, in the evaluated configuration, the maximum number of failed attempts is recommended to be set to 3.<br><br>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.<br><br>Administrator lockouts are not applicable to the local console. |
| 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 15 characters. |
| FIA_PSK_EXT.1 | Through the implementation of the CLI, the TOE supports use of 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 127 bytes in length. The data that is input is conditioned by the cryptographic module prior to use via SHA-1. |
| FIA_UIA_EXT.1<br>FIA_UAU_EXT.2 | The TOE requires all users to be successfully identified and authenticated before allowing any TSF mediated actions to be performed except for the login warning banner that is displayed prior to user authentication and any network packets as configured by the authorized administrator may flow through the switch.<br><br>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 CLI of the TOE through either a directly connected console or remotely through an SSHv2 secured 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.<br><br>Local password-based authentication for administrative users accessing the TOE through SSHv2, and optionally supports deferring password-based authentication to a remote AAA server. In addition, public key algorithm for authentication is RSA Signature Verification, which is configured during the TOE installation<br><br>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 then fail back to the local user database if the remote authentication servers are inaccessible. |

| TOE SFRs | How the SFR is Met |
|---|---|
| | The process for authentication is the same for administrative access whether administration is occurring via a directly connected console or remotely via SSHv2 secured connection.<br><br>At initial login, 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 grant administrative access (if the combination of username and password is correct) or indicate 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, the TOE displays only '*' characters so that the user password is obscured. For remote session authentication, the TOE does not echo any characters as they are entered. The TOE does not provide a reason for failure in the cases of a login failure. |
| FIA_X509_EXT.1/Rev<br>FIA_X509_EXT.2<br>FIA_X509_EXT.3 | The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for IPsec connections.<br><br>The TOE supports the following methods to obtain a certificate from a CA:<br>• Simple Certificate Enrolment Protocol (SCEP)—A Cisco-developed enrolment protocol that uses HTTP to communicate with the CA or registration authority (RA).<br>• Imports certificates in PKCS12 format from an external server<br>• IOS File System (IFS)—The switch uses any file system that is supported by Cisco IOS software (such as TFTP, FTP, flash, and NVRAM) to send a certificate request and to receive the issued certificate.<br>• Manual cut-and-paste—The switch displays the certificate request on the console terminal, allowing the administrator to enter the issued certificate on the console terminal; manually cut-and-paste certificate requests and certificates when there is no network connection between the switch and CA<br>• Enrolment profiles—The switch sends HTTP-based enrolment requests directly to the CA server instead of to the RA-mode certificate server (CS).<br>• Self-signed certificate enrolment for a trust point<br><br>All certificates include at least the following information: public key, Common Name, Organization, Organizational Unit and Country.<br><br>Public key infrastructure (PKI) credentials, such as RSA and ECDSA keys and certificates can be stored in a specific location on the TOE. 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. During run time, an authorized administrator can specify what active local storage device will be used to store certificates.<br><br>The certificates themselves provide protection in that they are digitally signed. If a certificate were modified in any way, it would be invalidated. The digital signature verifications process would show that the certificate has been tampered with and then the hash value would be invalid.<br><br>The certificate chain establishes a sequence of trusted certificates, from a peer certificate to the root CA certificate. Within the PKI hierarchy, all enrolled peers can validate the certificate of one another if the peers share a trusted root CA certificate or a common subordinate CA. Each CA corresponds to a trust point. When a certificate chain is received from a peer, the default processing of a certificate chain path continues until the first trusted certificate, or trust point, is reached. The administrator may configure the level to which a certificate chain is processed on all certificates including subordinate CA certificates.<br><br>To verify, the authorized administrator could 'show' the pki certificates and the pki trust points. |

| TOE SFRs | How the SFR is Met |
|---|---|
| | The authorized administrator can also configure one or more certificate fields together with their matching criteria to match. Such as:<br><br>• alt-subject-name<br>• expires-on<br>• issuer-name<br>• name<br>• serial-number<br>• subject-name<br>• unstructured-subject-name<br>• valid-start<br><br>This allows for installing more than one certificate from one or more CAs on the TOE. For example, one certificate from one CA could be used for one IPsec connection, while another certificate from another CA could be used for a different IPsec connection. However, the default configuration is a single certificate from one CA that is used for all authenticated connections.<br><br>The physical security of the TOE (A.PHYSICAL_PROTECTION) protects the switch and the certificates from being tampered with or deleted. Only authorized administrators with the necessary privilege level can access the certificate storage and add/delete them. In addition, the TOE identification and authentication security functions protect an unauthorized user from gaining access to the TOE.<br><br>CRL is used for certificate revocation checking. The authorized administrator could use the "revocation-check" command to specify at least one method of revocation checking; CRL is not the default method and must be selected in the evaluated configuration. The authorized administer sets the trust point and its name and the revocation-check method. If the TOE does not have the applicable CRL and is unable to obtain one, the TOE will reject the peer's certificate.<br><br>The extendedKeyUsage field is validated according to the following rules:<br><br>• Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.<br>• Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.<br>• Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.<br>OCSP certificats presented for OCSP responses shall have the OCSP signing purpose (id-kg 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field<br><br>Checking is also done for the basicConstraints extension and the CA flag to determine whether they are present and set to TRUE. The local certificate that was imported must contain the basic constraints extension with the CA flag set to true, the check also ensure that the key usage extension is present, and the keyEncipherment bit or the keyAgreement bit or both are set. If they are not, the certificate is not accepted.<br><br>The certificate chain path validation is configured on the TOE by first setting crypto pki trustpoint name and then configuring the level to which a certificate chain is processed on all certificates including subordinate CA certificates using the chain-validation command. If the connection to determine the certificate validity cannot be established, the certificate is not accepted, and the connection will not be established. |

| TOE SFRs | How the SFR is Met |
|---|---|
| | Note, certificate revocation check is performed when certificates are loaded on the device and on each use during the authentication step and is the same process for all certificates. |
| FMT_MOF.1/ManualUpdate<br>FMT_MOF.1/Functions<br>FMT_MOF.1/Services<br>FMT_MTD.1/CoreData<br>FMT_MTD.1/CryptoKeys | The TOE provides the ability for Security Administrators to access TOE data, such as audit data, configuration data, security attributes, routing tables, and session thresholds and to perform manual updates to the TOE. The Security Administrator has the ability to start and stop services. The Security Administrator enters the `crypto key generate` command in the CLI to generate RSA and ECDSA key pairs.<br><br>Each of the predefined and administratively configured roles has create (set), query, modify, or delete access to the TOE data. 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.<br><br>The term "Security 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. Semi-privileged administrators with only a subset of privileges can also modify TOE data based on the privileges granted. No administrative functionality is available prior to administrative login. |
| FMT_SMF.1<br>FMT_SMF.1/VPN | The TOE provides all the capabilities necessary to securely manage the TOE and the services provided by the TOE. The management functionality of the TOE is provided through the TOE CLI. The specific management capabilities available from the TOE include:<br><br>• Ability to administer the TOE locally and remotely;<br>• Ability to configure the access banner;<br>• Ability to configure the session inactivity time before session termination or locking;<br>• Ability to update the TOE, and to verify the updates using digital signature and [hash comparison] capability prior to installing those updates;<br>• Ability to configure the authentication failure parameters for FIA_AFL.1;<br>• Ability to manage the cryptographic keys;<br>• Ability to configure the cryptographic functionality;<br>• Ability to configure the lifetime for the IPsec SAs;<br>• Ability to import X.509v3 certificates to the TOE's trust store;<br>• Ability to start and stop services;<br>• Ability to configure audit behaviour (e.g., changes to storage locations for audit; changes to behaviour when local audit storage space is full);<br>• Ability to modify the behaviours of the transmission of audit data to an external IT entity, the handling of audit data, the audit functionality when Local Audit Storage Space is full;<br>• Ability to configure the list of TOE-provided services available before an entity is identified and authenticated as specified in FIA_UIA_EXT.1;<br>• Ability to configure thresholds for SSH rekeying;<br>• Ability to enable or disable automatic checking for updates or automatic updates;<br>• Ability to re-enable an Administrator account;<br>• Ability to set the time which is used for time-stamps;<br>• Ability to configure the reference identifier for the peer;<br>• Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors; |

| TOE SFRs | How the SFR is Met |
|---|---|
| | • Definition of packet filtering rules;<br>• Association of packet filtering rules to network interfaces;<br>• Ordering of packet filtering rules by priority<br>• Ability to manage the trusted public keys database;<br><br>Information about TSF-initiated Termination is covered in the TSS under FTA_SSL_EXT.1 or FTA_SSL.3. |
| 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.<br><br>The term "Security 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.<br><br>The privilege level determines the functions the user can perform; hence the Security Administrator with the appropriate privileges. Refer to the Guidance documentation and IOS Command Reference Guide for available commands and associated roles and privilege levels.<br><br>The TOE can and shall be configured to authenticate all access to the command line interface using a username and password.<br><br>The TOE supports both local administration via a directly connected console cable and remote administration via SSHv2. |
| FPF_RUL_EXT.1 | An authorized administrator can define the traffic that needs to be protected by configuring access lists (permit, deny, log) and applying these access lists to interfaces using access and crypto map sets. Therefore, traffic may be selected on the basis of the source and destination address, and optionally the Layer 4 protocol and port. The access lists can be applied to all the network interfaces.<br><br>The TOE enforces information flow policies on network packets that are received by TOE interfaces and leave the TOE through other TOE interfaces. 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.<br><br>By implementing rules that defines the permitted flow of traffic between interfaces of the TOE for unauthenticated traffic. These rules control whether a packet is transferred from one interface to another based on:<br><br>1. presumed address of source<br>2. presumed address of destination<br>3. transport layer protocol (or next header in IPv6)<br>4. Service used (UDP or TCP ports, both source and destination)<br>5. Network interface on which the connection request occurs |

| TOE SFRs | How the SFR is Met |
|---|---|
| | 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.<br><br>The TOE is capable of inspecting network packet header fields to determine if a packet is part of an established session or not. ACL rules still apply to packets that are part of an ongoing session.<br><br>Packets will be dropped unless a specific rule has been set up to allow the packet to pass (where the attributes of the packet match the attributes in the rule and the action associated with the rule is to pass traffic). This is the default action that occurs on an interface if no ACL rule is found. If a packet arrives that does not meet any rule, it is expected to be dropped. 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.<br><br>These rules are entered in the form of access lists at the CLI (via 'access list' and 'access group' commands). These interfaces reject traffic when the traffic arrives on an external TOE interface, and the source address is an external IT entity on an internal network;<br>These interfaces reject traffic when the traffic arrives on an internal TOE interface, and the source address is an external IT entity on the external network;<br>These interfaces reject traffic when the traffic arrives on either an internal or external TOE interface, and the source address is an external IT entity on a broadcast network;<br>These interfaces reject traffic when the traffic arrives on either an internal or external TOE interface, and the source address is an external IT entity on the loopback network;<br>These interfaces reject requests in which the subject specifies the route for information to flow when it is in route to its destination; and<br><br>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.<br><br>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. The initialization process first initializes the operating system, and then the networking daemons including the access list enforcement, prior to any daemons or user applications that potentially send network traffic. No incoming network traffic can be received before the access list functionality is operational. |
| FPT_FLS.1/SelfTest | 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 information to or from the TOE. The TOE shuts down by reloading and will continue to reload as long as the failures persist. This functionally prevents any failure of power-on self-tests, failure of integrity check of the TSF executable image, failure of noise source health tests from causing an unauthorized information flow. There are no failures that circumvent this protection. |

| TOE SFRs | How the SFR is Met |
|---|---|
| FPT_SKP_EXT.1<br>FPT_APW_EXT.1 | The TOE stores all private keys in a secure directory protected from access as there is no interface in which the keys can be accessed.<br><br>The TOE includes CLI command features 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. The password is encrypted by using the command "password encryption aes" used in global configuration mode.<br><br>The command *service password-encryption* applies encryption to all passwords, including username passwords, authentication key passwords, the privileged command password, console, and virtual terminal line access passwords. All passwords are stored encrypted in the configuration file.<br><br>Additionally, enabling the 'hidekeys' command in the logging configuration ensures that passwords are not displayed in plaintext.<br><br>The TOE includes a Master Passphrase feature that can be used to configure the TOE to encrypt all locally defined user passwords using AES. The Master Passphrase is set by using the key config-key password-encryption command. The Master Passphrase is stored in a secure directory that cannot be viewed by an Administrator. If the Master Passphrase is lost, it cannot be recovered. The Master Passphrase can be overwritten by creating a new passphrase or deleted by the Administrator.<br><br>In this manner, the TOE ensures that plaintext user passwords will not be disclosed even to administrators. Password encryption is configured using the 'service password-encryption' command. There are no administrative interfaces available that allow passwords to be viewed as they are encrypted via the password-encryption service. |
| FPT_STM_EXT.1 | The TOE provides a source of date and time information used in audit event timestamps, and for certificate validity checking. The clock function is reliant on the system clock provided by the underlying hardware. 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. The time information is also used in various routing protocols such as, OSPF, BGP, and ERF; Set system time, Calculate IKE stats (including limiting SAs based on times); determining AAA timeout, and administrative session timeout. |
| FPT_TUD_EXT.1 | An Authorized Administrator can query the software version running on the TOE and can initiate updates to software images. The current active version can be verified by executing the "show version" command from the TOE's CLI. When software updates are made available by Cisco, an administrator can obtain, verify the integrity of, and install those updates. The updates can be downloaded from the software.cisco.com.<br><br>The cryptographic hashes (i.e., SHA-512) are used to verify software 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. Authorized Administrators can download the approved image file from Cisco.com onto a trusted computer system for usage in the trusted update functionality. The hash value can be displayed by hovering over the software image name under details on the Cisco.com web site. The verification should not be performed on the TOE during the update process. If the hashes do not match, contact Cisco Technical Assistance Center (TAC).<br><br>Digital signatures and published hash mechanisms 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. The TOE image files are digitally signed so their integrity can be verified during the boot process, and an image that fails an integrity check will not be loaded. |

| TOE SFRs | How the SFR is Met |
| --- | --- |
| | To verify the digital signature prior to installation, the “show software authenticity file” command allows you to display software authentication related information that includes image credential information, key type used for verification, signing information, and other attributes in the signature envelope, for a specific image file. If the output from the “show software authenticity file” command does not provide the expected output, contact Cisco Technical Assistance Center (TAC) https://tools.cisco.com/ServiceRequestTool/create/launch.do.<br><br>Further instructions for how to do this verification are provided in the administrator guidance for this evaluation.<br><br>Software images are available from Cisco.com at the following:<br>http://www.cisco.com/cisco/software/navigator.html |
| FPT_TST_EXT.1<br>FPT_TST_EXT.3 | The TOE runs a suite of self-tests during initial start-up to verify its correct operation. Refer to the FIPS Security Policy for available options and management of the cryptographic self-test. For testing of the TSF, the TOE automatically runs checks and tests at startup and during resets and periodically during normal operation to ensure the TOE is operating correctly, including checks of image integrity and all cryptographic functionality.<br><br>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:<br><br>• AES Known Answer Test –<br>For the encrypt test, a known key is used to encrypt a known plain text value resulting in an encrypted value. This encrypted value is compared to a known encrypted value to ensure that the encrypt operation is working correctly. The decrypt test is just the opposite. In this test a known key is used to decrypt a known encrypted value. The resulting plaintext value is compared to a known plaintext value to ensure that the decrypt operation is working correctly.<br><br>• RSA Signature Known Answer Test (both signature/verification) –<br>This test takes a known plaintext value and Private/Public key pair and used the public key to encrypt the data. This value is compared to a known encrypted value to verify that encrypt operation is working properly. The encrypted data is then decrypted using the private key. This value is compared to the original plaintext value to ensure the decrypt operation is working properly.<br><br>• RNG/DRBG Known Answer Test –<br>For this test, known seed values are provided to the DRBG implementation. The DRBG uses these values to generate random bits. These random bits are compared to known random bits to ensure that the DRBG is operating correctly.<br><br>• HMAC Known Answer Test –<br>For each of the hash values listed, the HMAC implementation is fed known plaintext data and a known key. These values are used to generate a MAC. This MAC is compared to a known MAC to verify that the HMAC and hash operations are operating correctly.<br><br>• SHA-1/256/384/512 Known Answer Test –<br>For each of the values listed, the SHA implementation is fed known data. These values are used to generate a hash. This hash is compared to a known value to verify they match, and the hash operations are operating correctly.<br><br>• ECDSA self-test –<br>This test takes a known plaintext value and Private/Public key pair and used the public key to encrypt the data. This value is compared to a known encrypted value to verify that encrypt |

| TOE SFRs | How the SFR is Met |
| --- | --- |
| | operation is working properly. The encrypted data is then decrypted using the private key. This value is compared to the original plaintext value to ensure the decrypt operation is working properly.<br><br>• Software Integrity Test –<br>The Software Integrity Test is run automatically whenever the IOS system image is loaded and confirms that the image file that's about to be loaded has maintained its integrity.<br><br>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.<br><br>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.<br><br>If an error occurs during the self-test, a SELF_TEST_FAILURE system log is generated.<br><br>Example Error Message _FIPS-2-SELF_TEST_IOS_FAILURE: "IOS crypto FIPS self test failed at %s."<br>Explanation FIPS self test on IOS crypto routine failed.<br><br>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 because any deviation in the TSF behavior will be identified by the failure of a self-test.<br><br>The integrity of stored TSF executable code when it is loaded for execution can be verified through the use of RSA and Elliptic Curve Digital Signature algorithms. |
| FTA_SSL_EXT.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. 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. If a remote user session is inactive for a configured period of time, the session will be terminated and will require authentication to establish a new session. |
| FTA_SSL.3<br>FTA_SSL.4 | An administrator is able to exit out of both local and remote administrative sessions. Each administrator logged onto the TOE can manually terminate their session using the "exit" or "logout" command.<br><br>The allowable inactivity timeout range is from 1 to 65535 seconds. Administratively configurable timeouts are also available for the EXEC level access (access above level 1) through use of the "exec-timeout" setting. |
| 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 interface is applicable for both local (via console) and remote (via SSH) TOE administration. |
| FTP_ITC.1<br>FTP_ITC.1/VPN | The TOE protects communications with peer or neighbor routers using keyed hash as defined in FCS_COP.1/KeyedHash and cryptographic hashing functions FCS_COP.1/Hash. 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/DataEncryption is provided to ensure the data is not disclosed in transit. The TSF allows the TSF, or the authorized IT entities to initiate communication via the trusted channel. |

| TOE SFRs | How the SFR is Met |
| --- | --- |
| | 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. The TOE also requires that peers establish an IKE/IPsec connection to a CA server for sending certificate signing requests.<br><br>The TOE protects communications between the TOE and the remote audit server using IPsec. This provides a secure channel to transmit the log events. Communications between the TOE and AAA servers are also secured using IPsec.<br><br>The distinction between "remote VPN 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 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 /Admin | 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. |

# 7 Key Zeroization

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

**Table 21 TOE Key Zeroization**

| Name | Description of Key | Zeroization |
|---|---|---|
| Diffie-Hellman Shared Secret | This is the shared secret used as part of the Diffie-Hellman key exchange. This key is stored in SDRAM. | Automatically after completion of DH exchange.<br><br>Overwritten with: 0x00 |
| Diffie Hellman private exponent | This is the private exponent used as part of the Diffie-Hellman key exchange. This key is stored in SDRAM. | Zeroized upon completion of DH exchange.<br><br>Overwritten with: 0x00 |
| Skeyid | This is an IKE intermittent value used to create skeyid_d. This key is stored in SDRAM. | Automatically after IKE session terminated.<br><br>Overwritten with: 0x00 |
| skeyid_d | This is an IKE intermittent value used to derive keying data for IPsec. This key is stored in SDRAM. | Automatically after IKE session terminated.<br><br>Overwritten with: 0x00 |
| IKE session encrypt key | This the key IPsec key used for encrypting the traffic in an IPsec connection. This key is stored in SDRAM. | Automatically after IKE session terminated.<br><br>Overwritten with: 0x00 |
| IKE session authentication key | This the key IPsec key used for authenticating the traffic in an IPsec connection. This key is stored in SDRAM. | Automatically after IKE session terminated.<br><br>Overwritten with: 0x00 |
| ISAKMP preshared | This is the configured pre-shared key for ISAKMP negotiation. This key is stored in NVRAM. | Zeroized using the following command:<br><br>**# no crypto isakmp key**<br><br>Overwritten with: 0x0d |
| IKE ECDSA Private Key | The ECDSA private-public key pair is created by the device itself using the key generation CLI command.<br><br>Afterwards, the device's public key must be put into the device certificate. The device's certificate is created by creating a trustpoint on the device. This trustpoint authenticates with the CA server to get the CA certificate and also enrolls with the CA server to generate the device certificate. | Zeroized using the following command:<br><br>**# crypto key zeroize ecdsa**[2]<br><br>Overwritten with: 0x0d |

[2] Issuing this command will zeroize/delete all ECDSA keys on the TOE.

| Name | Description of Key | Zeroization |
|---|---|---|
| | In the IKE authentication step, the device's certificate is firstly sent to another device to be authenticated. The other device verifies that the certificate is signed by CA's signing key, then sends back a random secret encrypted by the device's public key in the valid device certificate. . Only the device with the matching device private key can decrypt the message and obtain the random secret. This key is stored in NVRAM. | |
| IKE RSA Private Key | The RSA private-public key pair is created by the device itself using the key generation CLI described below.<br>Afterwards, the device's public key must be put into the device certificate. The device's certificate is created by creating a trustpoint on the device. This trustpoint authenticates with the CA server to get the CA certificate and also enrolls with the CA server to generate the device certificate.<br><br>In the IKE authentication step, the device's certificate is firstly sent to another device to be authenticated. The other device verifies that the certificate is signed by CA's signing key, then sends back a random secret encrypted by the device's public key in the valid device certificate. . Only the device with the matching device private key can decrypt the message and obtain the random secret. This key is stored in NVRAM. | Zeroized using the following command:<br><br>**# crypto key zeroize rsa**<br><br>Overwritten with: 0x0d |
| IPsec encryption key | This is the key used to encrypt IPsec sessions. This key is stored in SDRAM. | Automatically when IPsec session terminated.<br><br>Overwritten with: 0x00 |
| IPsec authentication key | This is the key used to authenticate IPsec sessions. This key is stored in SDRAM. | Automatically when IPsec session terminated.<br><br>Overwritten with: 0x00 |
| RADIUS secret | Shared secret used as part of the Radius authentication method. This key is stored in NVRAM. | Zeroized using the following command:<br><br>**# no radius-server key**<br><br>Overwritten with: 0x0d |
| SSH Private Key | Once the function has completed the operations requiring the RSA key object, the module overwrites the entire object (no matter its contents) using memset. This overwrites the key with all 0's. This key is stored in NVRAM. | Zeroized using the following command:<br>**# crypto key zeroize rsa**<br><br>Overwritten with: 0x00 |
| SSH Session Key | The results zeroized using the poisoning in free to overwrite the values with 0x00. This is called by the | Automatically when the SSH session is terminated. |

| Name | Description of Key | Zeroization |
| --- | --- | --- |
| | ssh_close function when a session is ended. This key is stored in SDRAM. | Overwritten with: 0x00 |

# Annex A: References

The following documentation was used to prepare this ST:

**Table 22 References**

| Identifier | Description |
|---|---|
| [CC_PART1] | Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and general model, dated April 2017, version 3.1, Revision 5 |
| [CC_PART2] | Common Criteria for Information Technology Security Evaluation – Part 2: Security functional components, dated April 2017, version 3.1, Revision 5 |
| [CC_PART3] | Common Criteria for Information Technology Security Evaluation – Part 3: Security assurance components, April 2017, version 3.1, Revision 5 |
| [CEM] | Common Methodology for Information Technology Security Evaluation – Evaluation Methodology, April 2017, version 3.1, Revision 5 |
| [NDcPP] | collaborative Protection Profile for Network Devices, v2.2e, 23 March 2020 |
| [MOD_VPNGW] | PP-Module for Virtual Private Network (VPN) Gateways (MOD_VPNGW), Version 1.0, November 22, 2019 |
| [800-38A] | NIST Special Publication 800-38A Recommendation for Block 2001 Edition Recommendation for Block Cipher Modes of Operation Methods and Techniques December 2001 |
| [800-56A] | NIST Special Publication 800-56A, March, 2007<br>Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography (Revised) |
| [800-56B] | NIST Special Publication 800-56B Recommendation for Pair-Wise, August 2009<br>Key Establishment Schemes Using Integer Factorization Cryptography |
| [FIPS 140-2] | FIPS PUB 140-2 Federal Information Processing Standards Publication<br>Security Requirements for Cryptographic Modules May 25, 2001 |
| [FIPS PUB 186-3] | FIPS PUB 186-3 Federal Information Processing Standards Publication Digital Signature Standard (DSS) June, 2009 |
| [FIPS PUB 186-4] | FIPS PUB 186-4 Federal Information Processing Standards Publication Digital Signature Standard (DSS) July 2013 |
| [FIPS PUB 198-1] | Federal Information Processing Standards Publication The Keyed-Hash Message Authentication Code (HMAC) July 2008 |
| [NIST SP 800-90A Rev 1] | NIST Special Publication 800-90A Recommendation for Random Number Generation Using Deterministic Random Bit Generators January 2015 |
| [FIPS PUB 180-3] | FIPS PUB 180-3 Federal Information Processing Standards Publication Secure Hash Standard (SHS) October 2008 |

## ANNEX B: TECHNICAL DECISIONS

This ST applies the following NIAP Technical Decisions:

**Table 23 NIAP Technical Decisions (TD)**

| TD Identifier | TD Name | Protection Profiles | References | Publication Date | Applicable? |
|---|---|---|---|---|---|
| TD0670 | NIT Technical Decision for Mutual and Non-Mutual Auth TLSC Testing | CPP_ND_V2.2E | ND SD2.2, FCS_TLSC_EXT.2.1 | 2022.08.26 | No, SFR not claimed |
| TD0639 | NIT Technical Decision for Clarification for NTP MAC Keys | CPP_ND_V2.2E | FCS_NTP_EXT.1.2, FAU_GEN.1, FCS_CKM.4, FPT_SKP_EXT.1 | 2022.08.26 | Yes |
| TD0638 | NIT Technical Decision for Key Pair Generation for Authentication | CPP_ND_V2.2E | NDSDv2.2, FCS_CKM.1 | 2022.08.05 | Yes |
| TD0636 | NIT Technical Decision for Clarification of Public Key User Authentication for SSH | CPP_ND_V2.2E | ND SD2.2, FCS_SSHC_EXT.1 | 2022.03.21 | No, SFR not claimed |
| TD0635 | NIT Technical Decision for TLS Server and Key Agreement Parameters | CPP_ND_V2.2E | FCS_TLSS_EXT.1.3, NDSD v2.2 | 2022.03.21 | No, SFR not claimed |
| TD0634 | NIT Technical Decision for Clarification required for testing IPv6 | CPP_ND_V2.2E | FCS_DTLSC_EXT.1.2, FCS_TLSC_EXT.1.2, ND SD v2.2 | 2022.03.21 | No, SFRs not claimed |
| TD0633 | NIT Technical Decision for IPsec IKE/SA Lifetimes Tolerance | CPP_ND_V2.2E | ND SD2.2, FCS_IPSEC_EXT.1.7, FCS_IPSEC_EXT.1.8 | 2022.03.21 | Yes |
| TD0632 | NIT Technical Decision for Consistency with Time Data for vNDs | CPP_ND_V2.2E | ND SD2.2, FPT_STM_EXT.1.2 | 2022.03.21 | No, not a virtual TOE |
| TD0631 | NIT Technical Decision for Clarification of public key authentication for SSH Server | CPP_ND_V2.2E | ND SDv2.2, FCS_SSHS_EXT.1, FMT_SMF.1 | 2022.03.21 | Yes |
| TD0597 | VPN GW IPv6 Protocol Support | MOD_VPNGW_v1.1 | FPF_RU_EXT.1.6 | 2021.08.26 | Yes |
| TD0592 | NIT Technical Decision for Local Storage of Audit Records | CPP_ND_V2.2E | FAU_STG | 2021.05.21 | Yes |
| TD0591 | NIT Technical Decision for Virtual TOEs and hypervisors | CPP_ND_V2.2E | A.LIMITED_FUNCTIONALITY, ACRONYMS | 2021.05.21 | No, not a virtual TOE |

| TD Identifier | TD Name | Protection Profiles | References | Publication Date | Applicable? |
|---|---|---|---|---|---|
| TD0590 | Mapping of operational environment objectives | MOD_VPNGW_v 1.1 | Section 4.3 | 2021.05.21 | Yes |
| TD0581 | NIT Technical Decision for Elliptic curve-based key establishment and NIST SP 800-56Arev3 | CPP_ND_V2.2E | FCS_CKM.2 | 2021.04.09 | Yes |
| TD0580 | NIT Technical Decision for clarification about use of DH14 in NDcPPv2.2e | CPP_ND_V2.2E | FCS_CKM.1.1, FCS_CKM.2.1 | 2021.04.09 | Yes |
| TD0572 | NiT Technical Decision for Restricting FTP_ITC.1 to only IP address identifiers | CPP_ND_V2.1, CPP_ND_V2.2E | FTP_ITC.1 | 2021.01.29 | Yes |
| TD0571 | NiT Technical Decision for Guidance on how to handle FIA_AFL.1 | CPP_ND_V2.1, CPP_ND_V2.2E | FIA_UAU.1, FIA_PMG_EXT.1 | 2021.01.29 | Yes |
| TD0570 | NiT Technical Decision for Clarification about FIA_AFL.1 | CPP_ND_V2.1, CPP_ND_V2.2E | FIA_AFL.1 | 2021.01.29 | Yes |
| TD0569 | NIT Technical Decision for Session ID Usage Conflict in FCS_DTLSS_EXT.1.7 | CPP_ND_V2.2E | ND SD v2.2, FCS_DTLSS_EXT.1.7 , FCS_TLSS_EXT.1.4 | 2021.01.28 | No, SFR not claimed |
| TD0564 | NiT Technical Decision for Vulnerability Analysis Search Criteria | CPP_ND_V2.2E | NDSDv2.2, AVA_VAN.1 | 2021.01.28 | Yes |
| TD0563 | NiT Technical Decision for Clarification of audit date information | CPP_ND_V2.2E | NDcPPv2.2e, FAU_GEN.1.2 | 2021.01.28 | Yes |
| TD0556 | NIT Technical Decision for RFC 5077 question | CPP_ND_V2.2E | NDSDv2.2, FCS_TLSS_EXT.1.4, Test 3 | 2020.11.06 | No, SFR not claimed |
| TD0555 | NIT Technical Decision for RFC Reference incorrect in TLSS Test | CPP_ND_V2.2E | NDSDv2.2, FCS_TLSS_EXT.1.4, Test 3 | 2020.11.06 | No, SFR not claimed |
| TD0549 | Consistency of Security Problem Definition update for MOD_VPNGW_v1.0 and MOD_VPNGW_v1.1 | MOD_VPNGW_v 1.0, MOD_VPNGW_v 1.1 | Section 6.1.2 | 2020.10.02 | Yes |
| TD0547 | NIT Technical Decision for Clarification on developer disclosure of AVA_VAN | CPP_ND_V2.1, CPP_ND_V2.2E | ND SDv2.1, ND SDv2.2, AVA_VAN.1 | 2020.10.15 | Yes |
| TD0546 | NIT Technical Decision for DTLS - clarification of Application Note 63 | CPP_ND_V2.2E | FCS_DTLSC_EXT.1.1 | 2020.10.15 | No, SFR not claimed |

| TD Identifier | TD Name | Protection Profiles | References | Publication Date | Applicable? |
|---|---|---|---|---|---|
| TD0538 | The NIT has issued a technical decision for Outdated link to allowed-with list | CPP_ND_V2.1, CPP_ND_V2.2E | Section 2 | 2020.07.13 | Yes |
| TD0537 | The NIT has issued a technical decision for Incorrect reference to FCS_TLSC_EXT.2.3 | CPP_ND_V2.2E | FIA_X509_EXT.2.2 | 2020.07.13 | No, SFR not claimed |
| TD0536 | The NIT has issued a technical decision for Update Verification Inconsistency | CPP_ND_V2.1, CPP_ND_V2.2E | AGD_OPE.1, ND SDv2.1, ND SDv2.2 | 2020.07.13 | Yes |
| TD0528 | The NIT has issued a technical decision for Missing EAs for FCS_NTP_EXT.1.4 | CPP_ND_V2.1, CPP_ND_V2.2E | FCS_NTP_EXT.1.4, ND SD v2.1, ND SD v2.2 | 2020.07.13 | No, SFR not claimed |
| TD0527 | Updates to Certificate Revocation Testing (FIA_X509_EXT.1) | CPP_ND_V2.2E | FIA_X509_EXT.1/REV, FIA_X509_EXT.1/ITT | 2020.07.01 | Yes |

# ANNEX C: EXTENDED COMPONENTS DEFINITIONS FOR NETWORK DEVICE COLABRATIVE PROTECT PROFILE

The NDcPPvNDcPP Author has defined extended components that are allowed by the cPP that are listed below and may be claimed in this Security Target (ST). Extended SFRs are identified by having a label "EXT" at the end of the Security Functional Requirement name.

Table 24 Extended Components

| Component Identification | Component Name |
|---|---|
| FAU_STG_EXT.1 | Protected Audit Event Storage |
| FCS_RBG_EXT.1 | Cryptographic Operation (Random Bit Generation) |
| FCS_IPSEC_EXT.1 | IPsec Protocol |
| FCS_SSHS_EXT.1 | SSH Server Protocol |
| FCS_RBG_EXT.1 | Cryptographic Operation (Random Bit Generation) |
| FIA_PMG_EXT.1 | Password Management |
| FIA_UIA_EXT.1 | User Identification and Authentication |
| FIA_UAU_EXT.2 | Password-based Authentication Mechanism |
| FIA_X509_EXT.1/Rev | X.509 Certificate Validation |
| FIA_X509_EXT.2 | X.509 Certificate Authentication |
| FIA_X509_EXT.3 | X.509 Certificate Requests |
| FPT_APW_EXT.1 | Protection of Administrator Passwords |
| FPT_SKP_EXT.1 | Protection of TSF Data (for reading of all pre-shared, symmetric and private keys) |
| FPT_STM_EXT.1 | Reliable Time Stamps |
| FPT_TST_EXT.1 | TSF Testing |
| FPT_TUD_EXT.1 | Trusted Update |
| FTA_SSL_EXT.1 | TSF-initiated Session Locking |

## Security Audit (FAU)

### Protected audit event storage (FAU_STG_EXT)

Family Behaviour

This component defines the requirements for the TSF to be able to securely transmit audit data between the TOE and an external IT entity.

Component levelling

FAU_STG_EXT.1 Protected audit event storage requires the TSF to use a trusted channel implementing a secure protocol.


FAU_STG_EXT.2 Counting lost audit data requires the TSF to provide information about audit records affected when the audit log becomes full.

FAU_STG_EXT.3 Action in case of possible audit data loss requires the TSF to generate a warning before the audit trail exceeds the local storage capacity.

FAU_STG_EXT.4 Protected Local audit event storage for distributed TOEs requires the TSF to use a trusted channel to protect audit transfer to another TOE component.

FAU_STG_EXT.5 Protected Remote audit event storage for distributed TOEs requires the TSF to use a trusted channel to protect audit transfer to another TOE component.

Management: FAU_STG_EXT.1, FAU_STG_EXT.2, FAU_STG_EXT.3, FAU_STG_EXT.4, FAU_STG_EXT.5

The following actions could be considered for the management functions in FMT:

a) The TSF shall have the ability to configure the cryptographic functionality.

Audit: FAU_STG_EXT.1, FAU_STG_EXT.2, FAU_STG_EXT.3, FAU_STG_EXT.4. FAU_STG_EXT.5

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) No audit necessary.

### A.1.1.1 FAU_ STG_EXT.1 Protected Audit Event Storage

| FAU_STG_EXT.1 | Protected Audit Event Storage |
| --- | --- |

Hierarchical to: No other components.
Dependencies: FAU_GEN.1 Audit data generation
FTP_ITC.1 Inter-TSF Trusted Channel

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 according to FTP_ITC.1

FAU_STG_EXT.1.2 The TSF shall be able to store generated audit data on the TOE itself. In addition [selection:

- *The TOE shall consist of a single standalone component that stores audit data locally,*
- *The TOE shall be a distributed TOE that stores audit data on the following TOE components: [assignment: identification of TOE components],*
- *The TOE shall be a distributed TOE with storage of audit data provided externally for the following TOE components: [assignment: list of TOE components that do not store audit data locally and the other TOE components to which they transmit their generated audit data]*.

FAU_STG_EXT.1.3 The TSF shall [selection: *drop new audit data, overwrite previous audit records according to the following rule: [assignment: rule for overwriting previous audit records], [assignment: other action]*] when the local storage space for audit data is full.

## Cryptographic Support (FCS)

### Random Bit Generation (FCS_RBG_EXT)

#### FCS_RBG_EXT.1 Random Bit Generation

Family Behaviour

Components in this family address the requirements for random bit/number generation. This is a new family defined for the FCS class.

Component levelling


FCS_RBG_EXT.1 Random Bit Generation requires random bit generation to be performed in accordance with selected standards and seeded by an entropy source.

Management: FCS_RBG_EXT.1

The following actions could be considered for the management functions in FMT:

a) There are no management activities foreseen

Audit: FCS_RBG_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) Minimal: failure of the randomization process

| FCS_RBG_EXT.1 | Random Bit Generation |
|---|---|

Hierarchical to: No other components
Dependencies: No other components

FCS_RBG_EXT.1.1 The TSF shall perform all deterministic random bit generation services in accordance with ISO/IEC 18031:2011 using [selection: *Hash_DRBG (any), HMAC_DRBG (any), CTR_DRBG (AES)*].

FCS_RBG_EXT.1.2 The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [selection: *[assignment: number of software-based sources] software-based noise source, [assignment: number of platform-based sources] platform-based noise source*] with a minimum of [selection: *128 bits, 192 bits, 256 bits*] of entropy at least equal to the greatest security strength, according to ISO/IEC 18031:2011 Table C.1 "Security Strength Table for Hash Functions", of the keys and hashes that it will generate.

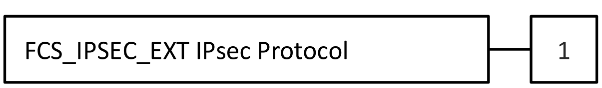
## Cryptographic Protocols (FCS_DTLSC_EXT, FCS_DTLSS_EXT, FCS_HTTPS_EXT, FCS_IPSEC_EXT, FCS_NTP_EXT, FCS_SSHC_EXT, FCS_SSHS_EXT, FCS_TLSC_EXT, FCS_TLSS_EXT)

### FCS_IPSEC_EXT.1 IPsec Protocol

Family Behaviour

Components in this family address the requirements for protecting communications using IPsec.

Component levelling




FCS_IPSEC_EXT.1 IPsec requires that IPsec be implemented as specified.

Management: FCS_IPSEC_EXT.1

The following actions could be considered for the management functions in FMT:

a) Maintenance of SA lifetime configuration

Audit: FCS_IPSEC_EXT.1

The following actions should be considered for audit if FAU_GEN Security audit data generation is included in the PP/ST:

a) Decisions to DISCARD, BYPASS, PROTECT network packets processed by the TOE.
b) Failure to establish an IPsec SA
c) IPsec SA establishment
d) IPsec SA termination
e) Negotiation "down" from an IKEv2 to IKEv1 exchange.

| FCS_IPSEC_EXT.1 | Internet Protocol Security (IPsec) Communications |
|---|---|

Hierarchical to: No other components

Dependencies: FCS_CKM.1 Cryptographic Key Generation
FCS_CKM.2 Cryptographic Key Establishment
FCS_COP.1/DataEncryption Cryptographic operation (AES Data encryption/decryption)
FCS_COP.1/SigGen Cryptographic operation (Signature Generation and Verification)
FCS_COP.1/Hash Cryptographic operation (Hash Algorithm)
FCS_COP.1/KeyedHash Cryptographic operation (Keyed Hash Algorithm)
FCS_RBG_EXT.1 Random Bit Generation

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

FCS_IPSEC_EXT.1.2 The TSF shall have a nominal, final entry in the SPD that matches anything that is otherwise unmatched and discards it.

FCS_IPSEC_EXT.1.3 The TSF shall implement [selection: *tunnel mode, transport mode*].

FCS_IPSEC_EXT.1.4 The TSF shall implement the IPsec protocol ESP as defined by RFC 4303 using the cryptographic algorithms [selection: *AES-CBC-128 (RFC 3602), AES-CBC-192 (RFC 3602), AES-CBC-256 (RFC 3602), AES-GCM-128 (RFC 4106), AES-GCM-192 (RFC 4106), AES-GCM-256 (RFC 4106),*] together with a Secure Hash Algorithm (SHA)-based HMAC [selection: *HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512, no HMAC algorithm*].

FCS_IPSEC_EXT.1.5 The TSF shall implement the protocol: [selection:

- *IKEv1, using Main Mode for Phase 1 exchanges, as defined in RFCs 2407, 2408, 2409, RFC 4109,* [selection: *no other RFCs for extended sequence numbers, RFC 4304 for extended sequence numbers*]*, and* [selection: *no other RFCs for hash functions, RFC 4868 for hash functions*]*;*

- *IKEv2 as defined in RFCs 5996 [selection: with no support for NAT traversal, with mandatory support for NAT traversal as specified in RFC 5996, section 2.23)], and* [selection: *no other RFCs for hash functions, RFC 4868 for hash functions*]].

FCS_IPSEC_EXT.1.6 The TSF shall ensure the encrypted payload in the [selection: *IKEv1, IKEv2*] protocol uses the cryptographic algorithms [selection: *AES-CBC-128, AES_CBC-192 AES-CBC-256 (specified in RFC 3602), AES-GCM-128, AES-GCM-192, AES-GCM-256 (specified in RFC 5282)*].

FCS_IPSEC_EXT.1.7 The TSF shall ensure that [selection:

- *IKEv1 Phase 1 SA lifetimes can be configured by a Security Administrator based on* [selection:
  - *number of bytes;*
  - *length of time, where the time values can be configured within [assignment: integer range including 24*] *hours;*

  ]*;*
- *IKEv2 SA lifetimes can be configured by a Security Administrator based on* [selection:
  - *number of bytes;*
  - *length of time, where the time values can be configured within [assignment: integer range including 24*] *hours*

  ]

].

FCS_IPSEC_EXT.1.8 The TSF shall ensure that [selection:

- *IKEv1 Phase 2 SA lifetimes can be configured by a Security Administrator based on* [selection:
  - *number of bytes;*
  - *length of time, where the time values can be configured within [assignment: integer range including 8] hours;*

  ]*;*
- *IKEv2 Child SA lifetimes can be configured by a Security Administrator based on* [selection:
  - *number of bytes;*
  - *length of time, where the time values can be configured within [assignment: integer range including 8] hours;*

  ]

].

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 *[assignment: (one or more) number(s) of bits that is at least twice the security strength of the negotiated Diffie-Hellman group*] bits.

FCS_IPSEC_EXT.1.10 The TSF shall generate nonces used in [selection: *IKEv1, IKEv2*] exchanges of length [selection:

- *according to the security strength associated with the negotiated Diffie-Hellman group;*
- *at least 128 bits in size and at least half the output size of the negotiated pseudorandom function (PRF) hash*

  ].

FCS_IPSEC_EXT.1.11 The TSF shall ensure that IKE protocols implement DH Group(s) [selection:

- [selection*: 14 (2048-bit MODP), 15 (3072-bit MODP), 16 (4096-bit MODP), 17 (6144-bit MODP), 18 (8192-bit MODP)] according to RFC 3526,*
- [selection*: 19 (256-bit Random ECP), 20 (384-bit Random ECP), 21 (521-bit Random ECP), 24 (2048-bit MODP with 256-bit POS)] according to RFC 5114.*

].
FCS_IPSEC_EXT.1.12 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 [selection: *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 [selection: *IKEv1 Phase 2, IKEv2 CHILD_SA*] connection.

FCS_IPSEC_EXT.1.13 The TSF shall ensure that all IKE protocols perform peer authentication using [selection: *RSA, ECDSA*] that use X.509v3 certificates that conform to RFC 4945 and [selection: *Pre-shared Keys, no other method*].

FCS_IPSEC_EXT.1.14 The TSF shall only establish a trusted channel if the presented identifier in the received certificate matches the configured reference identifier, where the presented and reference identifiers are of the following fields and types: [selection: *SAN: IP address, SAN: Fully Qualified Domain Name (FQDN), SAN: user FQDN, CN: IP address, CN: Fully Qualified Domain Name (FQDN), CN: user FQDN, Distinguished Name (DN)]* and [selection: *no other reference identifier type, [assignment: other supported reference identifier types*]].

## FCS_SSHS_EXT.1 SSH Server Protocol

Family Behaviour

The component in this family addresses the ability for a server to offer SSH to protect data between a client and the server using the SSH protocol.

Component levelling


FCS_SSHS_EXT.1 SSH Server requires that the server side of SSH be implemented as specified.

Management: FCS_SSHS_EXT.1

The following actions could be considered for the management functions in FMT:

a) There are no management activities foreseen.

Audit: FCS_SSHS_EXT.1

The following actions should be considered for audit if FAU_GEN Security audit data generation is included in the PP/ST:

a) Failure of SSH session establishment
b) SSH session establishment
c) SSH session termination

| FCS_SSHS_EXT.1 | SSH Server Protocol |
|---|---|

Hierarchical to: No other components

Dependencies: FCS_CKM.1Cryptographic Key Generation
FCS_CKM.2 Cryptographic Key Establishment
FCS_COP.1/DataEncryption Cryptographic operation (AES Data encryption/decryption)
FCS_COP.1/SigGen Cryptographic operation (Signature Generation and Verification)
FCS_COP.1/Hash Cryptographic operation (Hash Algorithm)
FCS_COP.1/KeyedHash Cryptographic operation (Keyed Hash Algorithm)
FCS_RBG_EXT.1 Random Bit Generation

FCS_SSHS_EXT.1.1 The TSF shall implement the SSH protocol in accordance with: RFCs 4251, 4252, 4253, 4254, [selection: *4256, 4344, 5647, 5656, 6187, 6668, 8268, 8308* section 3.1, 8332].
FCS_SSHS_EXT.1.2 The TSF shall ensure that the SSH protocol implementation supports the following user authentication methods as described in RFC 4252: public key-based, [selection: *password-based, no other method*].

FCS_SSHS_EXT.1.3 The TSF shall ensure that, as described in RFC 4253, packets greater than *[assignment: number of bytes]* bytes in an SSH transport connection are dropped.

FCS_SSHS_EXT.1.4 The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms and rejects all other encryption algorithms: *[assignment: encryption algorithms]*.

FCS_SSHS_EXT.1.5 The TSF shall ensure that the SSH public-key based authentication implementation uses [selection: *ssh-rsa, rsa-sha2-256, rsa-sha2-512, ecdsa-sha2-nistp256, x509v3-ssh-rsa, ecdsa-sha2-nistp384, ecdsa-sha2-nistp521, x509v3-ecdsa-sha2-nistp256, x509v3-ecdsa-sha2-nistp384, x509v3-ecdsa-sha2-nistp521, x509v3-rsa2048-sha256*] as its public key algorithm(s) and rejects all other public key algorithms.

FCS_SSHS_EXT.1.6 The TSF shall ensure that the SSH transport implementation uses *[assignment: list of MAC algorithms]* as its MAC algorithm(s) and rejects all other MAC algorithm(s).

FCS_SSHS_EXT.1.7 The TSF shall ensure that *[assignment: list of key exchange methods]* are the only allowed key exchange methods used for the SSH protocol.

FCS_SSHS_EXT.1.8 The TSF shall ensure that within SSH connections, the same session keys are used for a threshold of no longer than one hour, and each encryption key is used to protect no more than one gigabyte of data. After any of the thresholds are reached, a rekey needs to be performed.

## Identification and Authentication (FIA)

### Password Management (FIA_PMG_EXT)

Family Behaviour

The TOE defines the attributes of passwords used by administrative users to ensure that strong passwords and passphrases can be chosen and maintained.
Component levelling


FIA_PMG_EXT.1 Password management requires the TSF to support passwords with varying composition requirements, minimum lengths, maximum lifetime, and similarity constraints.

Management: FIA_PMG_EXT.1

No management functions.

Audit: FIA_PMG_EXT.1

No specific audit requirements.

#### A.1.1.2 FIA_PMG_EXT.1 Password Management

| | |
|---|---|
| FIA_PMG_EXT.1 | Password Management |

Hierarchical to: No other components.
Dependencies: No other components.

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

*a)* Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters: [selection: *“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, “)”, [assignment: other characters]*];

*b)* Minimum password length shall be configurable to between [*assignment: minimum number of characters supported by the TOE*] and [*assignment: number of characters greater than or equal to 15*] characters.

### User Identification and Authentication (FIA_UIA_EXT)

Family Behaviour

The TSF allows certain specified actions before the non-TOE entity goes through the identification and authentication process.

Component levelling


FIA_UIA_EXT.1 User Identification and Authentication requires Administrators (including remote Administrators) to be identified and authenticated by the TOE, providing assurance for that end of the communication path. It also ensures that every user is identified and authenticated before the TOE performs any mediated functions

Management: FIA_UIA_EXT.1

The following actions could be considered for the management functions in FMT:

a) Ability to configure the list of TOE services available before an entity is identified and authenticated

Audit: FIA_UIA_EXT.N

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) All use of the identification and authentication mechanism
b) Provided user identity, origin of the attempt (e.g. IP address)

### A.1.1.3 FIA_UIA_EXT.1 User Identification and Authentication

| FIA_UIA_EXT.1 | User Identification and Authentication |
|---|---|

Hierarchical to: No other components.
Dependencies: FTA_TAB.1 Default TOE Access Banners

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;
- [selection: *no other actions, automated generation of cryptographic keys, [assignment: list of services, actions performed by the TSF in response to non-TOE requests]*].

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

### User authentication (FIA_UAU_EXT)

Family Behaviour
Provides for a locally based administrative user authentication mechanism

Component levelling


FIA_UAU_EXT.2 The password-based authentication mechanism provides administrative users a locally based authentication mechanism.

Management: FIA_UAU_EXT.2

The following actions could be considered for the management functions in FMT:

a) None

Audit: FIA_UAU_EXT.2

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) Minimal: All use of the authentication mechanism

### A.1.1.4 FIA_UAU_EXT.2 Password-based Authentication Mechanism

| FIA_UAU_EXT.2 | Password-based Authentication Mechanism |
| --- | --- |

Hierarchical to: No other components.
Dependencies: No other components.

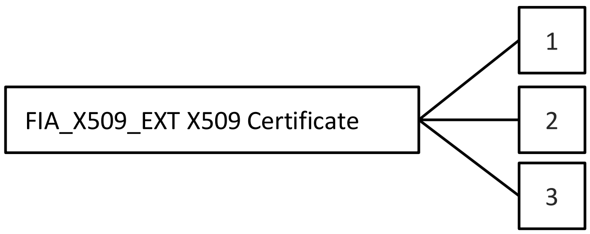
FIA_UAU_EXT.2.1 The TSF shall provide a local [selection: *password-based, SSH public key-based, certificate-based, [assignment: other authentication mechanism(s)]*] authentication mechanism to perform local administrative user authentication.

**Authentication using X.509 certificates (FIA_X509_EXT)**

Family Behaviour

This family defines the behaviour, management, and use of X.509 certificates for functions to be performed by the TSF. Components in this family require validation of certificates according to a specified set of rules, use of certificates for authentication for protocols and integrity verification, and the generation of certificate requests.

Component levelling




FIA_X509_EXT.1 X509 Certificate Validation, requires the TSF to check and validate certificates in accordance with the RFCs and rules specified in the component.

FIA_X509_EXT.2 X509 Certificate Authentication, requires the TSF to use certificates to authenticate peers in protocols that support certificates, as well as for integrity verification and potentially other functions that require certificates.

FIA_X509_EXT.3 X509 Certificate Requests, requires the TSF to be able to generate Certificate Request Messages and validate responses.

Management: FIA_X509_EXT.1, FIA_X509_EXT.2, FIA_X509_EXT.3

The following actions could be considered for the management functions in FMT:

a) Remove imported X.509v3 certificates
b) Approve import and removal of X.509v3 certificates
c) Initiate certificate requests

Audit: FIA_X509_EXT.1, FIA_X509_EXT.2, FIA_X509_EXT.3

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) Minimal: No specific audit requirements are specified.

### A.1.1.5FIA_X509_EXT.1 X.509 Certificate Validation

| FIA_X509_EXT.1 | X.509 Certificate Validation |
|---|---|

Hierarchical to: No other components
Dependencies: FIA_X509_EXT.2 X.509 Certificate Authentication

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

- RFC 5280 certificate validation and certification path validation.
- The certification path must terminate with a trusted CA certificate designated as a trust anchor.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of the certificate using [selection: *the Online Certificate Status Protocol (OCSP) as specified in RFC 6960, a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3, Certificate Revocation List (CRL) as specified in RFC 5759 Section 5, no revocation method*]
- The TSF shall validate the extendedKeyUsage field according to the following rules: *[assignment: rules that govern contents of the extendedKeyUsage field that need to be verified]*.

FIA_X509_EXT.1.2 The TSF shall only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.

#### A.1.1.6FIA_X509_EXT.2 X509 Certificate Authentication

| FIA_X509_EXT.2 | X.509 Certificate Authentication |
|---|---|

Hierarchical to: No other components
Dependencies: FIA_X509_EXT.1 X.509 Certificate Validation

FIA_X509_EXT.2.1 The TSF shall use X.509v3 certificates as defined by RFC 5280 to support authentication for [selection*: DTLS, HTTPS, IPsec, TLS, SSH, [assignment: other protocols], no protocols*], and [selection: *code signing for system software updates [assignment: other uses], no additional uses*].

FIA_X509_EXT.2.2 When the TSF cannot establish a connection to determine the validity of a certificate, the TSF shall [selection: *allow the Administrator to choose whether to accept the certificate in these cases, accept the certificate, not accept the certificate*].

#### A.1.1.7FIA_X509_EXT.3 X.509 Certificate Requests

| FIA_X509_EXT.3 | X.509 Certificate Requests |
|---|---|

Hierarchical to: No other components
Dependencies: FCS_CKM.1 Cryptographic Key Generation
FIA_X509_EXT.1 X.509 Certificate Validation

FIA_X509_EXT.3.1 The TSF shall generate a Certificate Request as specified by RFC 2986 and be able to provide the following information in the request: public key and [selection: *device-specific* *information, Common Name, Organization, Organizational Unit, Country, [assignment: other information]*].

FIA_X509_EXT.3.2 The TSF shall validate the chain of certificates from the Root CA upon receiving the CA Certificate Response.

## Protection of the TSF (FPT)

### Protection of TSF Data (FPT_SKP_EXT)

Family Behaviour

Components in this family address the requirements for managing and protecting TSF data, such as cryptographic keys. This is a new family modelled after the FPT_PTD Class.

Component levelling


FPT_SKP_EXT.1 Protection of TSF Data (for reading all symmetric keys), requires preventing symmetric keys from being read by any user or subject. It is the only component of this family.

Management: FPT_SKP_EXT.1

The following actions could be considered for the management functions in FMT:

a) There are no management activities foreseen.

Audit: FPT_SKP_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) There are no auditable events foreseen.

### A.1.1.8 FPT_SKP_EXT.1 Protection of TSF Data (for reading of all symmetric keys)

| FPT_SKP_EXT.1 | Protection of TSF Data (for reading of all symmetric keys) |
|---|---|

Hierarchical to: No other components.
Dependencies: No other components.

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

### FPT_APW_EXT.1 Protection of Administrator Passwords

Family Behaviour

Components in this family ensure that the TSF will protect plaintext credential data such as passwords from unauthorized disclosure.

Component levelling

FPT_APW_EXT Protection of Administrator Passwords — 1

FPT_APW_EXT.1 Protection of Administrator passwords requires that the TSF prevent plaintext credential data from being read by any user or subject.

Management: FPT_APW_EXT.1

The following actions could be considered for the management functions in FMT:

a) No management functions.

Audit: FPT_APW_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) No audit necessary.

| FPT_APW_EXT.1 | Protection of Administrator Passwords |
|---|---|

Hierarchical to: No other components
Dependencies: No other components.

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

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

## TSF Self-Test (FPT_TST_EXT)

Family Behaviour

Components in this family address the requirements for self-testing the TSF for selected correct operation.

Component levelling


FPT_TST_EXT.1 TSF Self-Test requires a suite of self-tests to be run during initial start-up in order to demonstrate correct operation of the TSF.

Management: FPT_TST_EXT.1

The following actions could be considered for the management functions in FMT:

a) No management functions.

Audit: FPT_TST_EXT.1

The following actions should be considered for audit if FAU_GEN Security audit data generation is included in the PP/ST:

a) Indication that TSF self-test was completed
b) Failure of self-test

| | |
|---|---|
| FPT_TST_EXT.1 | TSF Testing |

Hierarchical to: No other components.
Dependencies: No other components.

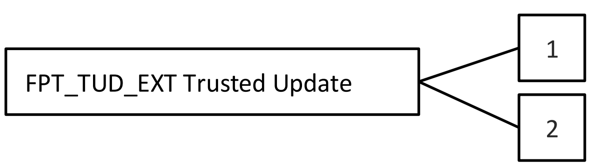
FPT_TST_EXT.1.1 The TSF shall run a suite of the following self-tests [selection: *during initial start-up (on power on), periodically during normal operation, at the request of the authorised user, at the conditions [assignment: conditions under which self-tests should occur]*] to demonstrate the correct operation of the TSF: *[assignment: list of self-tests run by the TSF*].

## Trusted Update (FPT_TUD_EXT)

Family Behaviour

Components in this family address the requirements for updating the TOE firmware and/or software.

Component levelling



FPT_TUD_EXT.1 Trusted Update requires management tools be provided to update the TOE firmware and software, including the ability to verify the updates prior to installation.

FPT_TUD_EXT.2 Trusted update based on certificates applies when using certificates as part of trusted update and requires that the update does not install if a certificate is invalid.

Management: FPT_TUD_EXT.1, FPT_TUD_EXT.2

The following actions could be considered for the management functions in FMT:

a) Ability to update the TOE and to verify the updates
b) Ability to update the TOE and to verify the updates using the digital signature capability (FCS_COP.1/SigGen) and [selection: *no other functions, [assignment: other cryptographic functions (or other functions) used to support the update capability]*]
c) Ability to update the TOE, and to verify the updates using [selection: *digital signature, published hash, no other mechanism*] capability prior to installing those updates

Audit: FPT_TUD_EXT.1, FPT_TUD_EXT.2

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) Initiation of the update process.
b) Any failure to verify the integrity of the update

### A.1.1.9 FPT_TUD_EXT.1 Trusted Update

| | |
|---|---|
| FPT_TUD_EXT.1 | Trusted Update |

Hierarchical to: No other components

Dependencies: FCS_COP.1/SigGen Cryptographic operation (for Cryptographic Signature and Verification), orFCS_COP.1/Hash Cryptographic operation (for cryptographic hashing)

FPT_TUD_EXT.1.1 The TSF shall provide [assignment: *Administrators*] the ability to query the currently executing version of the TOE firmware/software and [selection: *the most recently installed version of the TOE firmware/software; no other TOE firmware/software version*].

FPT_TUD_EXT.1.2 The TSF shall provide [assignment: *Administrators*] the ability to manually initiate updates to TOE firmware/software and [selection: *support automatic checking for updates, support automatic updates, no other update mechanism*].

FPT_TUD_EXT.1.3 The TSF shall provide means to authenticate firmware/software updates to the TOE using a [selection: *X.509 certificate, digital signature, published hash*] prior to installing those updates.

### Time stamps (FPT_STM_EXT)

Family Behaviour

Components in this family extend FPT_STM requirements by describing the source of time used in timestamps.

Component levelling


FPT_STM_EXT.1 Reliable Time Stamps is hierarchic to FPT_STM.1: it requires that the TSF provide reliable time stamps for TSF and identifies the source of the time used in those timestamps.

Management: FPT_STM_EXT.1

The following actions could be considered for the management functions in FMT:

a) Management of the time
b) Administrator setting of the time.

Audit: FTA_SSL_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) Discontinuous changes to the time.

## A.1.1.10 FPT_STM_EXT.1 Reliable Time Stamps

| | |
|---|---|
| FPT_STM_EXT.1 | Reliable Time Stamps |

Hierarchical to: No other components
Dependencies: No other components.

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

FPT_STM_EXT.1.2 The TSF shall [selection: *allow the Security Administrator to set the time, synchronise time with an NTP server*].

# TOE Access (FTA)

## TSF-initiated Session Locking (FTA_SSL_EXT)

Family Behaviour

Components in this family address the requirements for TSF-initiated and user-initiated locking, unlocking, and termination of interactive sessions.
The extended FTA_SSL_EXT family is based on the FTA_SSL family.

Component levelling


FTA_SSL_EXT.1 TSF-initiated session locking, requires system initiated locking of an interactive session after a specified period of inactivity. It is the only component of this family.

Management: FTA_SSL_EXT.1

The following actions could be considered for the management functions in FMT:

c) Specification of the time of user inactivity after which lock-out occurs for an individual user.

Audit: FTA_SSL_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

b) Any attempts at unlocking an interactive session.

### A.1.1.11 FTA_SSL_EXT.1 TSF-initiated Session Locking

| FTA_SSL_EXT.1 | TSF-initiated Session Locking |
| --- | --- |

Hierarchical to: No other components
Dependencies: FIA_UAU.1 Timing of authentication

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

- *lock the session - disable any activity of the Administrator's data access/display devices other than unlocking the session, and requiring that the Administrator re-authenticate to the TSF prior to unlocking the session;*

- *terminate the session*]

after a Security Administrator-specified time period of inactivity.

# ANNEX D: EXTENDED COMPONENTS DEFINITIONS FOR PP-MODULE FOR VIRTUAL PRIVATE NETWORK (VPN) GATEWAYS

The VPN Gateway Module Author has defined extended components that are allowed by the PP-Module, that are listed below and may be claimed in this Security Target (ST). Extended SFRs are identified by having a label "EXT" at the end of the Security Functional Requirement name.

Table 25 Extended Components

| Component Identification | Component Name |
|---|---|
| FIA_PSK_EXT.1 | Pre-Shared Key Composition |
| PRF_RUL_EXT.1 | Packet Filtering |
| FPT_TSF_EXT.1 | Protection of the TSF |
| FTA_VCM_EXT.1 | VPN Client Management |

## FIA: Identification and Authentication

### FIA_PSK_EXT Pre-Shared Key Compostion

#### 1.9.1.1 Family Behavior

This family defines requirements for what the TSF defines or generates as an acceptably strong pre-shared key for authentication.

#### 1.9.1.2 Component Leveling

| FIA_PSK_EXT Pre-Shared Key Composition | 1 |
|---|---|

FIA_PSK_EXT.1, Pre-Shared Key Composition, requires the TSF to use only pre-shared keys that meetcertain strength requirements.

#### 1.9.1.3 Management: FIA_PSK_EXT.1

No specific management functions are identified.

#### 1.9.1.4 Audit: FIA_PSK_EXT.1

There are no auditable events foreseen.

FIA_PSK_EXT.1 Pre-Shared Key Composition

Hierarchical to: No other components

Dependencies: FCS_COP.1 Cryptographic Operation

FIA_PSK_EXT.1.1 The TSF shall be able to use pre-shared keys for IPsec and [*selection: no other protocols, [assignment: other protocols that use pre-shared keys]*].

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

- Are 22 characters and [*selection: [assignment: other supported lengths], no other lengths*];
- composed of any combination of upper and lower case letters, numbers, andspecial characters (that include: "!", "@", "#", "$", "%", "^", "&", "*", "(", and")").

FIA_PSK_EXT.1.3 The TSF shall condition the text-based pre-shared keys by using [*selection: SHA-1, SHA-256, SHA-512, [assignment: method of conditioning text string]*].

FIA_PSK_EXT.1.4 The TSF shall be able to [*selection: accept, generate (using the random bitgenerator specified in FCS_RBG_EXT.1)*] bit-based pre-shared keys.

## FPF: Packet Filtering

This class contains families that describe packet filtering behavior. Packet filtering refers to the notion that network traffic that is transmitted "through" the TOE (i.e. the source and destination of the traffic is not the TOE but the TOE is on the routing path between these two entities) can be treated differently by the TSF based on attributes associated with the traffic. As this class is defined solely to contain an extended component defined for this PP-Module, it only has one family, FPF_RUL_EXT.

### FPR_RUL_EXT Packet Filtering Rules

#### Family Behavior

This family defines the requirements for the rules that are used to perform packet filtering of networktraffic.

#### Component Leveling

| | |
|---|---|
| FPF_RUL_EXT Packet Filtering Rules | 1 |

FPF_RUL_EXT.1, Packet Filtering Rules, requires the TSF to enforce a given set of packet filtering rules in an administrator-defined order against one or more TSFIs.

#### Management: FPF_RUL_EXT.1

The following actions could be considered for the management functions in FMT:

- Ability to configure the TOE's packet filtering functionality (i.e. the operations to be performed on network traffic based on configured attributes, the interfaces that these are associated with, and theorder in which they are applied)

Audit: FPF_RUL_EXT.1

Following actions should be auditable if FAU_GEN Security audit data generation is included in thePP/ST:

- Application of rules configured with the 'log' operation (including source/destination address, source/destination port, and transport layer protocol value)

FPF_RUL_EXT.1 Packet Filtering Rules Hierarchical

to: No other components

Dependencies: No dependencies

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 allow the definition of Packet Filtering rules using the followingnetwork protocol fields:

- IPv4 (RFC 791)
  - Source address
  - Destination Address
  - Protocol
- IPv6 (RFC 2460)
  - Source address
  - Destination Address
  - Next Header (Protocol)
- TCP (RFC 793)
  - Source Port
  - Destination Port
- UDP (RFC 768)
  - Source Port
  - Destination Port

FPF_RUL_EXT.1.3 The TSF shall allow the following operations to be associated with Packet Filteringrules: permit, discard, and log.

FPF_RUL_EXT.1.4 The TSF shall allow the Packet Filtering rules to be assigned to each distinctnetwork interface.

FPF_RUL_EXT.1.5 The TSF shall process the applicable Packet Filtering rules (as determined inaccordance with FPF_RUL_EXT.1.4) in the following order: Administrator-defined.

FPF_RUL_EXT.1.6 The TSF shall drop traffic if a matching rule is not identified.

## FPT Protection of the TSF

### TSF Self-Test (FPT_TST_EXT)

This family is defined in the Base-PP. This PP-Module augments the extended family by adding one additional component, FPT_TST_EXT.3. This new component and its impact on the extended family's component leveling are shown below; reference the Base-PP for all other definitions for this family.

#### 1.9.1.5 Component Leveling

FPT_TST_EXT.3, TSF Self-Test with Defined Methods, requires the TSF to specify the method(s) by which self-testing is performed in addition to identifying the self-tests that are executed and the circumstances in which this execution occurs.

#### 1.9.1.6 Management: FPT_TST_EXT.3

No specific management functions are identified.

#### 1.9.1.7 Audit: FPT_TST_EXT.3

The following actions should be auditable if FAU_GEN Security audit data generation is included in thePP/ST:

- Indication that TSF self-test was completed
- Failure of self-test

FPT_TST_EXT.3 TSF Self-Test with Defined Methods

| | |
|---|---|
| Hierarchical to: | FPT_TST_EXT.1 TSF Self-Test |
| Dependencies: | No dependencies |
| FPT_TST_EXT.3.1 | The TSF shall run a suite of the following self-tests [*selection: during initial start- up (on power on), periodically during normal operation, at the request of the authorized user, at the conditions [assignment: conditions under which self-tests should occur]*] to demonstrate the correct operation of the TSF: [*assignment: listof self-tests run by the TSF*]. |
| FPT_TST_EXT.3.2 | The TSF shall execute the self-testing through [*assignment: method usedto evaluate the success or failure of self-testing*]. |

## FTA: TOE ACCESS

### FTA_VCM_EXT VPN Client Management

#### 1.9.1.1 Family Behavior

This family defines requirements for how the TSF interacts with VPN clients in its operational environment.

#### 1.9.1.2 Component Leveling

| FTA_VCM_EXT VPN Client Management | 1 |
|---|---|

FTA_VCM_EXT.1, VPN Client Management, requires the TSF to assign private (internal) IP addressesto VPN clients that successfully establish IPsec connections with it.

#### 1.9.1.3 Management: FTA_VCM_EXT.1

No specific management functions are identified.

#### 1.9.1.4 Audit: FTA_VCM_EXT.1

There are no auditable events foreseen.

#### 1.9.1.5 FTA_VCM_EXT.1 VPN Client Management

Hierarchical to: No other components
Dependencies: FCS_IPSEC_EXT.1 Internet Protocol Security (IPsec)
Communications [FTP_ITC.1 Inter-TSF Trusted Channel, or
FTP_TRP.1 Trusted Path]

FTA_VCM_EXT.1.1 The TSF shall assign a private IP address to a VPN client upon successfulestablishment of a security session.