Cisco Catalyst 8500 Series Edge Routers (Cat8500) running IOS-XE 17.18

Security Target

Version: 2.0
Date: October 10, 2025

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

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
ACL	Access Control Lists
AES	Advanced Encryption Standard
AGD	Guidance Document
AH	Authentication Header
BGP	Border Gateway Protocol
CA	Certificate Authority
CAVP	Cryptographic Algorithm Validation Program
CC	Common Criteria for Information Technology Security Evaluation
CEM	Common Evaluation Methodology for Information Technology Security
CLI	Command-Line Interface
CM	Configuration Management
CN	Common Name
CS	Certificate Server
CRL	Certificate Revocation List
CSfC	Commercial Solutions for Classified
CSR	Certificate Signing Request
DN	Distinguished Name
DRAM	Dynamic random-access memory
DSS	Digital Signature Standard
ESP	Encapsulating Security Payload
FIPS	Federal Information Processing Standards
FQDN	Fully Qualified Domain Name
GE	Gigabit Ethernet port
HTTPS	Hyper-Text Transport Protocol Secure
IC2M	IOS Common Cryptographic Module
IKE	Internet Key Exchange
IOS	Internetwork Operating System
IPsec	Internet Protocol (IP) secure (sec)
ISO	International Organization for Standardization
IT	Information Technology
LAN	Local Area Network
NDcPP	Network Device collaborative Protection Profile
NIAP	National Information Assurance Partnership
NIST	National Institute of Standards and Technology
NIT	Network Device iTC Interpretation Team
NTP	Network Time Protocol
NVRAM	Non-Volatile Random-Access Memory
OCSP	Online Certificate Status Protocol
OID	Object Identifier
OSP	Organizational Security Policies
OSPF	Open Shortest Path First
PoE	Power over Ethernet
POST	Power on Startup
PKI	Public Key Infrastructure
PP	Protection Profile
RBG	Random Bit Generator
RFC	Request For Comments

SA	Security Association
SAR	Security Assurance Requirements
SCEP	Simple Certificate Enrollment Protocol
SFP	Small–Form-factor Pluggable port
SFR	Security Functional Requirement
SHS	Secure Hash Standard
SKP	Signing Key Pair
SPD	Security Policy Definition
SSH	Secure Shell
SSL	Secure Sockets Layer
ST	Security Target
TAC	Technical Assistance Center
TCP	Transmission Control Protocol
TD	Technical Decision
TOE	Target of Evaluation
TSC	Target of Evaluation Security Function Scope of Control
TSF	Target of Evaluation Security Function
TSP	Target of Evaluation Security Policy
UDP	User Datagram Protocol
WAN	Wide Area Network
VPN	Virtual Private Network

Document Introduction

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

This document provides the basis for an evaluation of a specific Target of Evaluation (TOE), the Cisco Catalyst 8500 Series Edge Routers (Cat8500) running IOS-XE 17.18. This Security Target (ST) defines a set of assumptions about the aspects of the environment, a list of threats that the product intends to counter, a set of security objectives, a set of security requirements, and the IT security functions provided by the TOE which meet the set of requirements. Administrators of the TOE will be referred to as administrators, Authorized Administrators, TOE administrators, semi-privileged, privileged administrators, and security administrators in this document.

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]
- Annex A: Key Zeroization [Section 7]
- Annex B: References [Section 8]
- Annex C: NIAP Technical Decisions [Section 9]
- Annex D: Obtaining Documentation [Section 10]

The structure and content of this ST comply with the requirements specified in the Common Criteria (CC), Part 1, Annex A, and Part 2.

1.1 ST and TOE Reference

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

Table 2 ST and TOE Identification

Name	Description
ST Title	Cisco Catalyst 8500 Series Edge Routers (Cat8500) running IOS-XE 17.18 Security Target
ST Version	2.0
Publication Date	October 10, 2025
Vendor and ST Author	Cisco Systems, Inc.
TOE Reference	Cisco Catalyst 8500 Series Edge Routers (Cat8500) running IOS-XE 17.18
TOE Hardware Models	C8500-20X6C
TOE Software Version	IOS-XE 17.18
Keywords	Router, Network Appliance, Data Protection, Authentication, Cryptography, Secure Administration, Network Device, Virtual Private Network (VPN), VPN Gateway, MACsec

1.2 TOE Overview

The Cisco Catalyst 8500 Series Edge Routers (herein after referred to as the Cat8500 respectively) are purpose-built, routing platforms that include VPN functionality and MACsec encryption provided by the TOE hardware and the Cisco IOS-XE 17.18 software. The TOE includes the hardware models as defined in Table 4. This Security Target only addresses the functions that provide for the security of the TOE itself as described in Section 1.8 Logical Scope of the TOE. Functionality not described in the Protection Profile is outside the scope of the evaluation.

1.3 TOE Product Type

The TOE is a Network Device that includes VPN functionality and MACsec encryption as defined in NDcPPv3.0e, PKG_SSH_V1.0, MOD_VPNGW v1.3, and MOD_MACSEC_V1.0. The TOE is comprised of both software and hardware. The hardware is comprised of the Cat8500 routers as described in 1.7 Physical Scope of the TOE. The software is comprised of the Cisco Internet Operating System (IOS) XE software image Release IOS-XE 17.18.

The Cat8500 Series Edge routers are purpose built for high performance and integrated services. The Cat8500 provides higher WAN port density, redundant power supply capability and have options to choose from lower and higher module density. The Cat8500 provides IPsec connection capabilities to facilitate secure communications with external entities as required. The Cat8500 is a purpose-built, routing platform that includes VPN functionality and MACsec encryption.

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 3 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.
Management Workstation with IPSec Client	Yes	This includes any IT Environment Management workstation with an IPSec client installed that is used by the TOE administrator to support TOE administration over IPSec protected channels. Any IPSec client that supports IKEv2 or the appropriate IPSec protocols 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.
Certificate 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.
MACsec Peer	Yes	This includes any MACsec peer with which the TOE participates in MACsec communications. It may be any device that supports MACsec communications.
Remote VPN Gateway/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.
NTP Server	Yes	The TOE supports communications with an NTP Server in order to synchronize the date and time for a reliable timestamp on the TOE.

1.5 TOE Description

This section provides an overview of the Cat8500 Target of Evaluation (TOE). The TOE is comprised of both software and hardware. The hardware is comprised of the Cat8500 routers as described in 1.7 Physical Scope of the TOE. The TOE software for each platform is comprised of Cisco IOS-XE version 17.18. The Cisco IOS-XE version 17.18 software is used to meet all of the requirements as specified in this document regardless of the hardware platform.

1.5.1 Cisco Catalyst 8500 Series Edge Routers (Cat8500)

This section defines the Cat8500 components included in the evaluated configuration of the TOE. The TOE is comprised of both software and hardware. The software is comprised of Cisco IOS-XE version 17.18. The hardware models included in the evaluation are: C8500-20X6C. The TOE consists of several components including:

- Chassis: The TOE chassis includes 3RU form factor
- One Serial console port RJ45 or micro-USB
- Two USB Type A slots
- One management Gigabit Ethernet interface
- Integrated Gigabit Ethernet ports:
  - Twenty built-in 1/10GE SFP Ethernet ports
  - Six built-in 40/100GE Ethernet SFP ports
- Flash storage: 32GB
- DRAM: The default is 64GB.

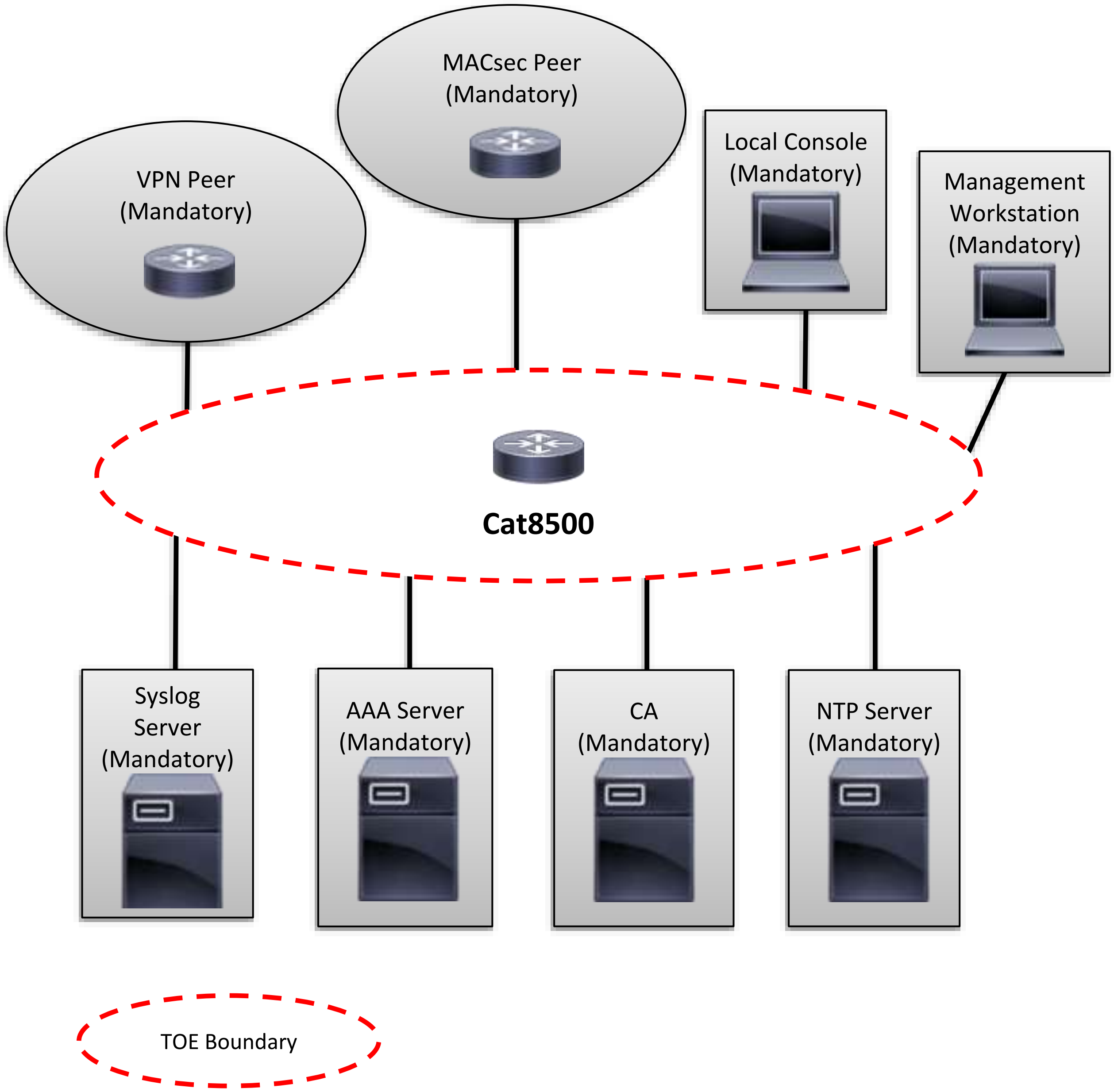
1.6 TOE Evaluated Configuration

1.6.1 Cisco Catalyst 8500 Series Edge Routers (Cat8500)

The TOE consists of one physical device as specified in section 1.7 below and includes Cisco IOS-XE version 17.18 software. The hardware model included in the evaluation is the C8500-20X6C. Table 4 adds additional details on the physical characteristics. The TOE has two or more network interfaces and is connected to at least one internal and one external network. The Cisco IOS-XE configuration determines how packets are handled to and from the TOE's network interfaces. The router configuration will determine how traffic flows received on an interface will be handled. Typically, packet flows are passed through the internetworking device and forwarded to their configured destination.

The following figure provides a visual depiction of an example Cat8500 TOE deployment:

Figure 1 TOE Example Deployment for Cat8500



The Figure 1 includes the following:

- Examples of TOE models
- The following are considered to be in the IT Environment:
  - VPN Peer
  - MACsec Peer
  - Management Workstation
  - Radius AAA (Authentication) Server
  - Audit (Syslog) Server
  - Local Console
  - Certificate Authority (CA)
  - NTP Server

NOTE: While the previous figure includes several non-TOE IT environment devices, the TOE is only the Cat8500 devices. Only one TOE device is required for deployment of the TOE in the 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:

- Cat8500
  - C8500-20X6C

The network, on which they reside, is considered part of the environment. The software is pre-installed and is comprised of the Cisco IOS-XE software image Release 17.18. In addition, the software image is 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 4 below:

Table 4 Hardware Models and Specifications

Hardware	Processor	Features
C8500-20X6C 	• Intel Xeon D-1573N (Broadwell) **MACsec** • Broadcom BCM82757 (1/10GE) • Comira MV88EC808 (40/100GE)	**Physical dimensions** 26.85” L x 17.25” W x 5.22” H 3RU **Interfaces** • 1x Serial console port RJ45 or micro-USB • 2x Type A USB • 1x 1GE Management port • 20x 1/10GE SFP ports with MACsec • 6x 40/100GE SFP ports with MACsec

1.8 Logical Scope of the TOE

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

- Security Audit
- Cryptographic Support
- Identification and Authentication

- Security Management
- Packet Filtering
- Protection of the TSF
- TOE Access
- Trusted Path/Channels

These features are described in more detail in the subsections below. In addition, the TOE implements all RFCs of the NDcPP v3.0e, PKG_SSH_V1.0, MOD_VPNGW v1.3 and MOD_MACSEC_V1.0 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 stored in persistent memory, with a configurable maximum size. When the log reaches the configured threshold, the TOE overwrites the oldest records. Audit logs are backed up over an encrypted channel to an external audit server.

1.8.2 Cryptographic Support

The TOE provides cryptography in support of other TOE security functionality. All the algorithms claimed have CAVP certificates for all processors listed in Table 4. The TOE leverages the IOS Common Cryptographic Module (IC2M) Rel5b (see Table 5 for certificate references). These cryptographic modules, including the MACsec cryptographic module, are implemented in a combination of hardware (such as the AES 4550 hardware implementation for MACsec) and firmware (such as the IC2M A4354 and C1668 implementations). The cryptographic operational environment for each of these modules is:

- Intel Xeon D-1573N (Broadwell) processor for the IOS Common Cryptographic Module (IC2M) Rel5b (A4354)
- Comira MV88EC808 for the GCM-AES Crypto Core (C1668)
- Broadcom BCM82757 for the AES ECB 128bit & 256bit Encryption/Decryption Engine (AES 4550)

Table 5 Cryptographic SFRs and Certificates

SFR	Algorithm	Description	Mode	Module	Certificate
FCS_COP.1/DataEncryption	AES	Symmetric Encryption/Decryption	AES-CBC-128 AES-CBC-192 AES-CBC-256 AES-GCM-128 AES-GCM-192 AES-GCM-256	IC2M	A4354
FCS_COP.1/Hash	SHA	Cryptographic Hashing Service	SHA-1 SHA-256 SHA-384 SHA-512	IC2M	A4354
FCS_COP.1/KeyedHash	HMAC	Keyed Hashing Service	HMAC-SHA-1 HMAC-SHA-256 HMAC-SHA-384 HMAC-SHA-512 implicit	IC2M	A4354
FCS_COP.1/CMAC	AES-CMAC	Keyed-Hash Message Authentication	AES-CMAC-128 AES-CMAC-256	IC2M	A4354
FCS_COP.1/MACSEC	AES-KW	Key Wrap	AES-KW-128 AES-KW-256	IC2M	A4354

SFR	Algorithm	Description	Mode	Module	Certificate
FCS_COP.1/MACSEC	AES-GCM	AEAD Symmetric Encryption/Decryption	AES-GCM-128 AES-GCM-256	MACsec	4550 C1668
FCS_COP.1/SigGen	RSA	RSA Signature Generation and Verification	PKCS #1v1.5 (2048, 3072)	IC2M	A4354
FCS_COP.1/SigGen	ECDSA	Elliptic Curve Signature Generation and Verification	P-256 P-384 P-521	IC2M	A4354
FCS_CKM.1 – Cryptographic Key Generation FCS_CKM.1/IKE – Cryptographic Key Generation	RSA	RSA Key Generation	2048 3072	IC2M	A4354
FCS_CKM.1 – Cryptographic Key Generation FCS_CKM.1/IKE – Cryptographic Key Generation	ECDSA	Elliptic Curve Key Generation	P-256 P-384	IC2M	A4354
FCS_CKM.1 – Cryptographic Key Generation FCS_CKM.1/IKE – Cryptographic Key Generation	FFC	FFC Key Generation	DH-14, DH-15, DH-16, DH-19, DH-20	IC2M	Tested by CCTL
FCS_CKM.2 – Cryptographic Key Establishment	KAS-ECC-SSC	ECC Key Establishment	P-256 P-384	IC2M	A4354
FCS_CKM.2 – Cryptographic Key Establishment	KAS-FFC-SSC	FFC Key Establishment	DH-14, DH-15, DH-16, DH-19, DH-20	IC2M	Tested by CCTL
FCS_RBG_EXT.1– Random Bit Generation	DRBG	Deterministic Random Bit Generation Services	CTR_DRBG (AES 256)	IC2M	A4354

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.

The cryptographic services provided by the TOE are described in Table 6 below:

Table 6 TOE Provided Cryptography

Cryptographic Method	Use within the TOE
Internet Key Exchange	Used to establish initial IPsec session.
Secure Shell Establishment	Used to establish initial SSH session.
RSA Signature Services	Used in IPsec session establishment. Used in SSH session establishment.
SP 800-90 RBG	Used in IPsec session establishment. Used in SSH session establishment. Used for random number generation, key generation and seeds to asymmetric key generation
SHS	Used to provide IPsec traffic integrity verification Used to provide SSH traffic integrity verification Used for keyed-hash message authentication

Cryptographic Method	Use within the TOE
AES	Used to encrypt IPsec session traffic. Used to encrypt SSH session traffic. Used to encrypt MACsec traffic Used to generate subkeys for CMAC
HMAC	Used for keyed hash, integrity services in IPsec and SSH session establishment.
RSA	Used in IKE protocols peer authentication Used to provide cryptographic signature services Used in Cryptographic Key Generation
ECDSA	Used to provide cryptographic signature services Used in Cryptographic Key Generation Used as the Key exchange method for IPsec
FFC DH	Used as the Key exchange method for IPsec
ECC DH	Used as the Key exchange method for SSH and IPsec

The TOE authenticates and encrypts packets between itself and a MACsec peer. The MACsec Key Agreement (MKA) Protocol provides the required session keys and manages the required encryption keys to protect data exchanged by the peers.

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 TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for IPsec connections. 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 fail exceeding the configured allowable attempts, the user is locked out until an authorized administrator can enable the user account.

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;
- NTP configurations;
- Update to the TOE and verification of the updates;
- Configuration of IPsec functionality.

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. 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.

The TOE is also capable of rejecting any MACsec PDUs in a given session that contain a SCI that is different from the one that is used to establish that session. The SCI is derived from the MACsec peer's MAC address and port to uniquely identify the originator of the MACsec PDU. Only EAPOL (PAE EtherType 88-8E), MACsec frames (EtherType 88-E5), MAC control frames EtherType 0x876F are permitted in the MACsec communication between peers and others are discarded.

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-XE is not a general-purpose operating system and access to Cisco IOS-XE memory space is restricted to only Cisco IOS-XE functions.

The TOE is also able to detect replay of information received via secure channels (MACsec). The detection applied to network packets that terminate at the TOE, such as trusted communications between the TOE and an IT entity (e.g., MACsec peer). If replay is detected, the packets are discarded.

The TOE has an internal clock; however, the TOE synchronizes time with an NTP server and then internally maintains the date and time. This date and time are used as the timestamp that is applied to audit records generated by the TOE.

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.

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

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 which has the ability to be encrypted further using IPsec and initiates outbound IPsec tunnels to transmit audit messages to remote syslog servers. The TOE can also establish trusted paths of peer-to-peer IPsec sessions. The peer-to-peer IPsec sessions can be used for securing the communications between the TOE and authentication server/syslog server, as well as to protect communications with a CA. In addition, IPsec is used to secure communications between the TOE and external entities, including remote authentication servers and NTP servers (via NTPv4 over IPsec).

1.9 Excluded Functionality

The following functionality is excluded from the evaluation:

Table 7 Excluded Functionality

Excluded Functionality	Exclusion Rationale
Non-FIPS 140-2 mode of operation	This mode of operation includes non-FIPS allowed operations.
USB ports	USB ports are not used for TOE functionality

These services will be disabled by configuration settings. The exclusion of this functionality does not affect compliance to the NDcPP v3.0e, MOD_VPNGW v1.3 and MOD_MACSEC_V1.0.

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 demonstrate exact conformance with the protection Profiles, Functional Packages, and Modules as listed below. This ST applies the NIAP Technical Decisions as described in Table 20 in Section 9.

- PP-Configuration for Network Devices, MACsec Ethernet Encryption, and VPN Gateways, Version 2.0 (CFG_NDcPP-MACsec-VPNGW_V2.0)
  - collaborative Protection Profile for Network Devices, Version 3.0e (CPP_ND_V3.0E)
  - PP-Module for Virtual Private Network (VPN) Gateways, Version 1.3 (MOD_VPNGW_V1.3)
  - PP-Module for MACsec Ethernet Encryption, Version 1.0 (MOD_MACSEC_V1.0)
- Functional Package for Secure Shell (SSH), Version 1.0 (PKG_SSH_V1.0)

2.3 Protection Profile Conformance Claim Rationale

2.3.1 TOE Appropriateness

The TOE provides all of the functionality at a level of security commensurate with that identified in the U.S. Government Protection Profile, PP-Module, and Functional Package:

- collaborative Protection Profile for Network Devices (NDcPP) Version 3.0e
- Functional Package for Secure Shell (SSH), Version 1.0
- PP-Module for Virtual Private Network (VPN) Gateways (MOD_VPNGW), Version 1.3
- PP-Module for MACsec Ethernet Encryption (MOD_MACSEC), Version 1.0

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 collaborative Protection Profile for Network Devices (NDcPP) Version 3.0e, Functional Package for Secure Shell (SSH), Version 1.0, PP-Module for Virtual Private Network (VPN) Gateway (MOD_VPNGW), Version 1.3 and PP-Module for MACsec Ethernet Encryption (MOD_MACSEC) Version 1.0 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 Version 3.0e, PKG_SSH_V1.0, MOD_VPNGW v1.3, and MOD_MACSEC_V1.0 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 v3.0e, PKG_SSH_V1.0, MOD_VPNGW v1.3 and MOD_MACSEC_V1.0 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 Version 3.0e, PKG_SSH_V1.0, MOD_VPNGW v1.3 and MOD_MACSEC_V1.0.

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.
- 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 8 TOE Assumptions

Assumption	Assumption Definition
A.CONNECTIONS	It is assumed that the TOE is connected to distinct networks in a manner that ensures that the TOE's security policies will be enforced on all applicable network traffic flowing among the attached networks.
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 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. For vNDs, this assumption applies to the physical platform on which the VM runs.
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 computing platform for general purpose applications (unrelated to networking functionality). If a virtual TOE evaluated as a pND, following Case 2 vNDs as specified in Section 1.2, the VS is considered part of the TOE with only one vND instance for each physical hardware platform. The exception being where components of a distributed TOE run inside more than one virtual machine (VM) on a single VS. In Case 2 vND, no non-TOE guest VMs are allowed on the platform.
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

Assumption	Assumption Definition
	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. 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.

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 9 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.

Threat	Threat Definition
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.
T.WEAK_AUTHENTICATION_ENDPOINTS	Threat agents may take advantage of secure protocols that use weak methods to authenticate the endpoints, e.g., shared 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. Nonvalidated 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

Threat	Threat Definition
	use by the attacker. Threat agents may also be able to take advantage of weak administrative passwords to gain privileged access to the device.
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.
T.DATA_INTEGRITY	Devices on a protected network may be exposed to threats presented by devices located outside the protected network that 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 communicate with those external devices then the data contained in the communications may be susceptible to a loss of integrity. An attacker may modify data transmitted over the layer 2 link in a way that is not detected by the recipient. Devices on a network may be exposed to attacks that attempt to corrupt or modify data in transit without authorization. If malicious devices are able to modify and replay data that is transmitted over a layer 2 link, then the data contained within the communications may be susceptible to a loss of integrity.
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. 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.

Threat	Threat Definition
	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. An attacker may send traffic through the TOE that enables them to access devices in the TOE's operational environment without authorization. A MACsec device may sit on the periphery of a network, which means that it may have an externally-facing interface to a public network. Devices located in the public network may attempt to exercise services located on the internal network that are intended to be accessed only from within the internal network or externally accessible only from specifically authorized devices. If the MACsec device allows unauthorized external devices access to the internal network, these devices on the internal network may be subject to compromise. Similarly, if two MACsec devices are deployed to facilitate end-to-end encryption of traffic that is contained within a single network, an attacker could use an insecure MACsec device as a method to access devices on a specific segment of that network such as an individual LAN.
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. 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

Threat	Threat Definition
	ports so that specific networks or network nodes can be blocked from accessing a protected network thereby further limiting the potential disclosure of information. 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 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.
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. 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. 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.
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:

Threat	Threat Definition
	- 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. - 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.
T.UNTRUSTED_MACSEC_COMMUNICATION_CHANNELS	An attacker may acquire sensitive TOE or user data that is transmitted to or from the TOE because an untrusted communication channel causes a disclosure of data in transit. A generic network device may be threatened by the use of insecure communications channels to transmit sensitive data. The attack surface of a MACsec device also includes the MACsec trusted channels. Inability to secure communications channels, or failure to do so correctly, would expose user data that is assumed to be secure to the threat of unauthorized disclosure.

3.3 Organizational Security Policies

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

Table 10 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 Administrators consent by accessing the TOE.

4 Security Objectives

This section 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.

Table 11 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) 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

TOE Objective	TOE Security Objective Definition
	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 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.
O.AUTHORIZED_ADMINISTRATION	All network devices are expected to provide services that allow the security functionality of the device to be managed. The MACsec device, as a specific type of network device, has a refined set of management functions to address its specialized behavior. In order to further mitigate the threat of a compromise of its security functionality, the MACsec device prescribes the ability to limit brute-force authentication attempts by enforcing lockout of accounts that experience excessive failures and by limiting access to security-relevant data that administrators do not need to view.
O.AUTHENTICATION_MACSEC	To further address the issues associated with unauthorized disclosure of information, a compliant TOE's authentication ability (MKA) will allow a MACsec peer to establish connectivity associations (CAs) with another MACsec peer. MACsec endpoints authenticate each other to ensure they are communicating with an authorized MAC Security Entity (SecY) entity.
O.CRYPTOGRAPHIC_FUNCTIONS_MACSEC	To address the issues associated with unauthorized modification and disclosure of information, compliant TOEs will implement 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.PORT_FILTERING_MACSEC	To further address the issues associated with unauthorized network access, a compliant TOE's port filtering capability will restrict the flow of network traffic through the TOE based on layer 2 frame characteristics and whether or not the traffic represents valid MACsec frames and MACsec Key Agreement Protocol Data Units (MKPDUs).
O.REPLAY_DETECTION	A MACsec device is expected to help mitigate the threat of MACsec data integrity violations by providing a mechanism to detect and discard replayed traffic for MPDUs.
O.SYSTEM_MONITORING_MACSEC	To address the issues of administrators being able to monitor the operations of the MACsec device, compliant TOEs will implement the ability to log the flow of Ethernet traffic. Specifically, the TOE will provide the means for administrators to configure rules to 'log' when Ethernet traffic grants or restricts access. As a result, the 'log' will

TOE Objective	TOE Security Objective Definition
	result in informative event logs whenever a match occurs. In addition, the establishment of security CAs is auditable, not only between MACsec devices, but also with MAC Security Key Agreement Entities (KaYs).
O.TSF_INTEGRITY	To mitigate the security risk that the MACsec device may fail during startup, it is required to fail-secure if any self-test failures occur during startup. This ensures that the device will only operate when it is in a known state.

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 12 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. Note: For vNDs the TOE includes only the contents of the its own VM, and does not include other VMs or the VS.
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. For vNDs, this includes the VS Administrator responsible for configuring the VMs that implement ND functionality. 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.

Environment Security Objective	IT Environment Security Objective Definition
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 *italicized and underlined text*
- Refinement: Indicated with **bold** text;
- Selection: Indicated with underlined text;
- Iteration: Indicated by adding a string starting with "/" (e.g., "FCS_COP.1/Hash") or appending the iteration number in parenthesis, e.g., (1), (2), (3).
- 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, MOD_VPNGW and MOD_MACSEC:

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

5.2 TOE Security Functional Requirements

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

Table 13 Security Functional Requirements

Class Name	Component Identification	Component Name
FAU: Security audit	FAU_GEN.1	Audit Data Generation
	FAU_GEN.1/MACSEC	Audit Data Generation (MACsec)
	FAU_GEN.1/VPN	Audit data generation (VPN Gateway)
	FAU_GEN.2	User identity association
	FAU_STG.1	Protected Audit Trail Storage
	FAU_STG_EXT.1	Protected Audit Event Storage
FCS: Cryptographic support	FCS_CKM.1	Cryptographic Key Generation
	FCS_CKM.1/IKE	Cryptographic Key Generation (for IKE peer authentication)
	FCS_CKM.2	Cryptographic Key Establishment
	FCS_CKM.4	Cryptographic Key Destruction

Class Name	Component Identification	Component Name
	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.1KeyedHash	Cryptographic Operation (Keyed Hash Algorithm)
	FCS_COP.1/CMAC	Cryptographic Operation (AES-CMAC Keyed Hash Algorithm)
	FCS_COP.1/MACSEC	Cryptographic Operation (MACsec AES Data Encryption and Decryption)
	FCS_IPSEC_EXT.1	Ipsec Protocol
	FCS_MACSEC_EXT.1	MACsec
	FCS_MACSEC_EXT.2	MACsec Integrity and Confidentiality
	FCS_MACSEC_EXT.3	MACsec Randomness
	FCS_MACSEC_EXT.4	MACsec Key Usage
	FCS_MKA_EXT.1	MACsec Key Agreement
	FCS_NTP_EXT.1	NTP Protocol
	FCS_SSH_EXT.1	SSH Protocol
	FCS_SSHS_EXT.1	SSH Protocol - Server
	FCS_RBG_EXT.1	Random Bit Generation
FIA: Identification and authentication	FIA_AFL.1	Authentication Failure Handling
	FIA_PMG_EXT.1	Password Management
	FIA_PSK_EXT.1 (1)	Pre-Shared Key Composition
	FIA_PSK_EXT.1 (2)	Pre-Shared Key Composition
	FIA_PSK_EXT.2	Generated Pre-Shared Keys
	FIA_UIA_EXT.1	User Identification and Authentication
	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

Class Name	Component Identification	Component Name
FMT: Security management		
	FMT_MOF.1/ManualUpdate	Trusted Update - Management of Security Functions Behaviour
	FMT_MOF.1/Services	Trusted Update - Management of Security Functions Behavior
	FMT_MOF.1/Functions	Management of Security Functions Behaviour
	FMT_MTD.1/CoreData	Management of TSF Data
	FMT_MTD.1/CryptoKeys	Management of TSF Data
	FMT_SMF.1	Specification of Management Functions
	FMT_SMF.1/MACSEC	Specification of Management Functions (MACsec)
	FMT_SMF.1/VPN	Specification of Management Functions
	FMT_SMR.2	Restrictions on security roles
FPF: Packet Filtering	FPF_RUL_EXT.1	Packet Filtering Rules
FPT: Protection of the TSF	FPT_APW_EXT.1	Protection of Administrator Passwords
	FPT_FLS.1	Failure with Preservation of Secure State
	FPT_FLS.1/SelfTest	Failure with Preservation of Secure State (Self-Test Failures)
	FPT_SKP_EXT.1	Protection of TSF Data (for reading of all symmetric keys)
	FPT_STM_EXT.1	Reliable Time Stamps
	FPT_TST_EXT.1	TSF Testing
	FPT_TST_EXT.3	Self-Test with Defined Methods
	FPT_TUD_EXT.1	Trusted Update
	FPT_RPL.1	Replay Detection
	FPT_RPL_EXT.1	Replay Protection for XPN
	FPT_CAK_EXT.1	Protection of CAK Data
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

Class Name	Component Identification	Component Name
FTP: Trusted path/channels	FTP_ITC.1	Inter-TSF trusted channel
	FTP_ITC.1/MACSEC	Inter-TSF Trusted Channel (MACsec Communications)
	FTP_ITC.1/VPN	Inter-TSF Trusted Channel (VPN Communications)
	FTP_TRP.1/Admin	Trusted Path

5.3 SFRs from NDcPP, MOD_VPNGW, MOD_MACsec, and PKG_SSH_V1.0

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 Administrator 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 Administrator account shall be logged), no other actions, [*Starting and stopping services*]*];*

*d) Specifically defined auditable events listed in Table **14**.*

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

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

b) For each audit event type, based on the auditable event definitions of the functional components included in the cPP/ST, *information specified in column three of Table **14**.*

5.3.1.2 FAU_GEN.1/VPN Audit Data Generation (VPN Gateway)

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

a) Start-up and shutdown of the audit functions

b) Indication that TSF self-test was completed

c) Failure of self-test

d) All auditable events for the [*not specified*] level of audit; and

e) [*auditable events defined in **the Auditable Events Table 14***].

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

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

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [*additional information defined in the* ***Auditable Events Table 14*** *for each auditable event, where applicable*].

5.3.1.3 FAU_GEN.1/MACSEC Audit Data Generation (MACSEC)

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

a) Start-up and shutdown of the audit functions
b) All auditable events for the [*not specified*] level of audit;
c) All administrative actions;
d) **[Specifically defined auditable events listed in the Auditable Events table (Table 14)**]

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

a) Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and
b) For each audit event type, based on the auditable event definitions of the functional components included in the PP-**Module**/ST, [*information specified in column three of table (Table* ***14****)*].

Table 14 Auditable Events

SFR	Auditable Event	Additional Audit Record Contents
FAU_GEN.1	None.	None.
FAU_GEN.1/MACSEC	None.	None.
FAU_GEN.1/VPN	No events specified	N/A
FAU_GEN.2	None.	None.
FAU_STG.1	None.	None.
FAU_STG_EXT.1	Configuration of local audit settings.	Identity of account making changes to the audit configuration.
FCS_CKM.1	None.	None.
FCS_CKM.1/IKE	No events specified	N/A
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_COP.1/CMAC	None.	None.
FCS_COP.1/MACSEC	None.	None.

SFR	Auditable Event	Additional Audit Record Contents
FCS_IPSEC_EXT.1	Failure to establish an IPsec SA.	Reason for failure.
FCS_MACSEC_EXT.1	Session establishment	Secure Channel Identifier (SCI)
FCS_MACSEC_EXT.2	None.	None.
FCS_MACSEC_EXT.3	Creation and update of SAK	Creation and update times
FCS_MACSEC_EXT.4	Creation of CA	Connectivity Association Key Names (CKNs)
FCS_MKA_EXT.1	None.	None.
FCS_NTP_EXT.1	• Configuration of a new time server. • Removal of configured time server.	Identity if new/removed time server.
FCS_SSH_EXT.1	[Failure to establish SSH connection] [Establishment of SSH connection] [Termination of SSH connection session] [Dropping of packet(s) outside defined size limits]	[Reason for failure and Non-TOE endpoint of attempted connection (IP Address)] [Non-TOE endpoint of attempted connection (IP Address)] [Non-TOE endpoint of attempted connection (IP Address)] [Packet size]
FCS_SSHS_EXT.1	No events specified.	N/A
FCS_RBG_EXT.1	None.	None.
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_PSK_EXT.1 (1)	None.	None.
FIA_PSK_EXT.1 (2)	No events specified	N/A
FIA_PSK_EXT.2	No events specified	N/A
FIA_UIA_EXT.1	All use of the identification and authentication mechanisms.	Origin of the attempt (e.g., IP address).
FIA_UAU.7	None.	None.
FIA_X509_EXT.1/Rev	• Unsuccessful attempt to validate a certificate • Any addition, replacement or removal of trust anchors in the TOE's trust store	• Reason for failure of certificate validation • Identification of certificates added, replaced or removed as trust anchor in the TOE's trust store
FIA_X509_EXT.2	None.	None.

SFR	Auditable Event	Additional Audit Record Contents
FIA_X509_EXT.3	None.	None.
FMT_MOF.1/ManualUpdate	Any attempt to initiate a manual update	None.
FMT_MOF.1/Services	None.	None.
FMT_MOF.1/Functions	None.	None.
FMT_MTD.1/CoreData	None.	None.
FMT_MTD.1/CryptoKeys	None.	None.
FMT_SMF.1	All management activities of TSF data.	None.
FMT_SMF.1/VPN	All administrative actions	No additional information.
FMT_SMR.2	None.	None.
FPF_RUL_EXT.1	Application of rules configured with the ‘log’ operation	• Source and destination addresses • Source and destination ports • Transport Layer Protocol
FPT_APW_EXT.1	None.	None.
FPT_FLS.1/SelfTest	No events specified	N/A
FPT_SKP_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. Origin of the attempt to change time for success and failure (e.g., IP address).
FPT_TST_EXT.1	None.	None.
FPT_TST_EXT.3	No events specified	N/A
FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure)	None.
FPT_RPL.1	Detected replay attempt	None.
FPT_RPL_EXT.1	None.	None.
FPT_CAK_EXT.1	None.	None.
FTA_SSL_EXT.1	The termination of a local session by the session lock.	None.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	None.

SFR	Auditable Event	Additional Audit Record Contents
FTA_SSL.4	The termination of an interactive session.	None.
FTA_TAB.1	None.	None.
FTP_ITC.1	• Initiation of the trusted channel. • Termination of the trusted channel. • Failure of the trusted channel functions.	• None • None • Reason for failure
FTP_ITC.1/VPN	• Initiation of the trusted channel. • Termination of the trusted channel. • Failure of the trusted channel functions.	• No additional information • No additional information • Identification of the initiator and target of failed trusted channel establishment attempt
FTP_TRP.1/Admin	• Initiation of the trusted path • Termination of the trusted path. • Failure of the trusted path functions.	• None. • None. • Reason for Failure

5.3.1.4 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.

5.3.1.5 FAU_STG.1 Protected Audit Trail Storage

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

FAU_STG.1.2 The TSF shall be able to prevent unauthorised modifications to the stored audit records in the audit trail.

5.3.1.6 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
[

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

FAU_STG_EXT.1.3 The TSF shall maintain a [buffer] of audit records in the event that an interruption of communication with the remote audit server occurs.

FAU_STG_EXT.1.4 The TSF shall be able to store [persistent] audit records locally with a minimum storage size of [*8192 to 2147483647 bytes*].

FAU_STG_EXT.1.5 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.

FAU_STG_EXT.1.6 The TSF shall provide the following mechanisms for administrative access to locally stored audit records [ability to view locally].

5.3.2 Cryptographic Support (FCS)

5.3.2.1 FCS_CKM.1 Cryptographic Key Generation

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, 3072-bit] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3 or FIPS PUB 186-5, "Digital Signature Standard (DSS)", A.1;
- ECC schemes using 'NIST curves' [P-256, P-384] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4, or FIPS PUB 186-5, "Digital Signature Standard (DSS)", Appendix A.2, or ISO/IEC 14888-3, "IT Security techniques - Digital signatures with appendix - Part 3: Discrete logarithm based mechanisms", Section 6.6.;
- 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 [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 specified cryptographic key generation algorithm:

[

- **FIPS PUB 186-5 "Digital Signature Standard (DSS)", Appendix B.3 for RSA schemes;**
- **FIPS PUB 186-5, "Digital Signature Standard (DSS)", Appendix B.4 for ECDSA schemes and implementing "NIST curves" P-384 and [P-256]**]
  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 [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

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 meet 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 [groups listed in 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, a new value of the key]];
- *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, [*0x0d*]];

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] and [no other]** mode and cryptographic key sizes **[128 bits, 256 bits] and [192 bits]** that meet the following: AES as specified in ISO 18033-3, **[CBC as specified in ISO 10116, GCM as specified in ISO 19772] and [no other standards]**.

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,
- Elliptic Curve Digital Signature Algorithm

]

and cryptographic key sizes [

- For RSA: [*2048, 3072 bits*],
- For ECDSA: [*256, 384, 521 bits*]

]

that meet the following: [

- For RSA schemes: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 5.5, using PKCS #1 v2.1 or FIPS PUB 186-5, "Digital Signature Standard (DSS)", Section 5.4 using PKCS #1 v2.2 Signature Schemes RSASSA-PSS and/or RSASSA-PKCS1v1_5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3,

- For ECDSA schemes implementing [P-256, P-384, P-521] curves that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 6 and Appendix D, Implementing "NIST Recommended curves"; or FIPS PUB 186-5, "Digital Signature Standard (DSS)", Section 6 and NIST SP 800-186 Section 3.2.1, Implementing Weierstrass curves; or ISO/IEC 14888-3, "IT Security techniques - Digital signatures with appendix - Part 3: Discrete logarithm based mechanisms", Section 6.6.

].

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/IEC 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-384, HMAC-SHA-512, implicit] and cryptographic key sizes [*160-bit, 256-bit, 384-bit, 512-bit*] **and message digest sizes [160, 256, 384, 512] bits** that meet the following: *ISO/IEC 9797-2:2011, Section 7 “MAC Algorithm 2”.*

5.3.2.9 FCS_COP.1/CMAC Cryptographic Operation (AES-CMAC Keyed Hash Algorithm)

FCS_COP.1.1/CMAC The TSF shall perform [*keyed-hash message authentication*] in accordance with a specified cryptographic algorithm [*AES-CMAC*] and cryptographic key sizes [**128, 256**] **bits and message digest size of 128 bits** that meets the following: [NIST SP 800-38B].

5.3.2.10 FCS_COP.1/MACSEC Cryptographic Operation (MACsec AES Data Encryption and Decryption)

FCS_COP.1.1/MACSEC The TSF shall perform [*encryption and decryption*] in accordance with a specified cryptographic algorithm [*AES used in AES Key Wrap, GCM]* and cryptographic key sizes [**128, 256**] **bits** that meets the following: [*AES as specified in ISO 18033-3, AES Key Wrap as specified in NIST SP 800-38F, GCM as specified in ISO 19772*].

5.3.2.11 FCS_IPSEC_EXT.1 IPSEC Protocol

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 [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 [***AES-CBC-128, AES-CBC-256 (specified in RFC 3602), AES-GCM-128, AES-GCM-256 (specified in RFC 4106)*] and [*AES-CBC-192 (specified in RFC 3602), AES-GCM-192 (specified in 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 7296 and [with mandatory support for NAT traversal as specified in RFC 7296, 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 between [1 hour] and [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 between [1 hour] and [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), 384 (for DH Group 20), 256 (for DH Group 15), and 256 (for DH Group 16)*] 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 Groups

- **19 (256-bit Random ECP), 20 (384-bit Random ECP) according to RFC 5114 and**
  [
- *[14 (2048-bit MODP), 15 (3072-bit MODP), 16 (4096-bit MODP)] according to RFC 3526*
- *[no other DH Groups] according to RFC 5114*

].

FCS_IPSEC_EXT.1.12 The TSF shall be able to ensure 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 ***[IKEv2]*** protocols perform peer authentication using [*RSA, ECDSA*] that use X.509v3 certificates that conform to RFC 4945 and [*Pre-shared Keys that conform to RFC 8784, 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: **Distinguished Name (DN),** [*no other reference identifier types*].

5.3.2.12 FCS_ MACSEC_EXT.1 MACsec

FCS_MACSEC_EXT.1.1 The TSF shall implement MACsec in accordance with IEEE Standard 802.1AE-2018.

FCS_MACSEC_EXT.1.2 The TSF shall derive a Secure Channel Identifier (SCI) from a peer's MAC address and port to uniquely identify the originator of an MPDU.

FCS_MACSEC_EXT.1.3 The TSF shall reject any MPDUs during a given session that contain an SCI other than the one used to establish that session.

FCS_MACSEC_EXT.1.4 The TSF shall permit only EAPOL (Port Access Entity (PAE) EtherType 88-8E), MACsec frames (EtherType 88-E5), and [[*EtherType 0x876F*] no other frame types] and shall discard others.

5.3.2.13 FCS_ MACSEC_EXT.2 MACsec Integrity and Confidentiality

FCS_MACSEC_EXT.2.1 The TOE shall implement MACsec with support for integrity protection with a confidentiality offset of [*0, 30, 50*].

FCS_MACSEC_EXT.2.2 The TSF shall provide assurance of the integrity of protocol data units (MPDUs) using an Integrity Check Value (ICV) derived with the SAK.

FCS_MACSEC_EXT.2.3 The TSF shall provide the ability to derive an Integrity Check Value Key (ICK) from a Connectivity Association Key (CAK) using a KDF.

5.3.2.14 FCS_ MACSEC_EXT.3 MACsec Randomness

FCS_MACSEC_EXT.3.1 The TSF shall generate unique Secure Association Keys (SAKs) using [*key derivation from Connectivity Association Key (CAK) per section 9.8.1 of IEEE 802.1X-2010*] such that the likelihood of a repeating SAK is no less than 1 in 2 to the power of the size of the generated key.

FCS_MACSEC_EXT.3.2 The TSF shall generate unique nonces for the derivation of SAKs using the TOE's random bit generator as specified by FCS_RBG_EXT.1.

5.3.2.15 FCS_ MACSEC_EXT.4 MACsec Key Usage

FCS_MACSEC_EXT.4.1 The TSF shall support peer authentication using pre-shared keys (PSKs) [*no other method*].

FCS_MACSEC_EXT.4.2 The TSF shall distribute SAKs between MACsec peers using AES key wrap as specified in FCS_COP.1/**MACSEC**.

FCS_MACSEC_EXT.4.3 The TSF shall support specifying a lifetime for CAKs.

FCS_MACSEC_EXT.4.4 The TSF shall associate Connectivity Association Key Names (CKNs) with SAKs that are defined by the KDF using the CAK as input data (per IEEE 802.1X-2010, Section 9.8.1).

FCS_MACSEC_EXT.4.5 The TSF shall associate CKNs with CAKs. The length of the CKN shall be an integer number of octets, between 1 and 32 (inclusive).

5.3.2.16 FCS_ MKA_EXT.1 MACsec Key Agreement

FCS_MKA_EXT.1.1 The TSF shall implement Key Agreement Protocol (MKA) in accordance with IEEE 802.1X-2010 and 802.1Xbx-2014.

FCS_MKA_EXT.1.2 The TSF shall provide assurance of the integrity of MKA protocol data units (MKPDUs) using an Integrity Check Value (ICV) derived from an Integrity Check Value Key (ICK).

FCS_MKA_EXT.1.3 The TSF shall provide the ability to derive an Integrity Check Value Key (ICK) from a CAK using a KDF.

FCS_MKA_EXT.1.4 The TSF shall enforce an MKA Lifetime Timeout limit of 6.0 seconds and [MKA Bounded Hello Timeout limit of 2 seconds].

FCS_MKA_EXT.1.5 The Key Server shall refresh a SAK when it expires. The Key Server shall distribute a SAK by [

- *pairwise CAKs that are PSKs*

].

FCS_MKA_EXT.1.6 The Key Server shall distribute a fresh SAK whenever a member is added to or removed from the live membership of the CA.

FCS_MKA_EXT.1.7 The TSF shall validate MKPDUs according to IEEE 802.1X-2010, Section 11.11.2. In particular, the TSF shall discard without further processing any MKPDUs to which any of the following conditions apply:

a) The destination address of the MKPDU was an individual address
b) The MKPDU is less than 32 octets long
c) The MKPDU comprises fewer octets than indicated by the Basic Parameter Set body length, as encoded in bits 4 through 1 of octet 3 and bits 8 through 1 of octet 4, plus 16 octets of ICV
d) The CAK Name is not recognized

If an MKPDU passes these tests, then the TSF will begin processing it as follows:

a) If the Algorithm Agility parameter identifies an algorithm that has been implemented by the receiver, the ICV shall be verified as specified in IEEE 802.1X-2010 Section 9.4.1.
b) If the Algorithm Agility parameter is unrecognized or not implemented by the receiver, its value can be recorded for diagnosis but the received MKPDU shall be discarded without further processing.

Each received MKPDU that is validated as specified in this clause and verified as specified in IEEE 802.1X-2010 Section 9.4.1 shall be decoded as specified in IEEE 802.1X-2010 Section 11.11.4.

5.3.2.17 FCS_NTP_EXT.1 NTP Protocol

FCS_NTP_EXT.1.1 The TSF shall use only the following NTP version(s) [NTP v4 (RFC 5905)].

FCS_NTP_EXT.1.2 The TSF shall update its system time using [

- [IPsec] to provide trusted communication between itself and an NTP time source.

].

FCS_NTP_EXT.1.3 The TSF shall not update NTP timestamp from broadcast and/or multicast addresses.

FCS_NTP_EXT.1.4 The TSF shall support configuration of at least three (3) NTP time sources in the Operational Environment.

5.3.2.18 FCS_SSH_EXT.1 SSH Protocol

FCS_SSH_EXT.1.1 The TOE shall implement *SSH* acting as a [*server*] in accordance with that complies with RFCs 4251, 4252, 4253, 4254*,* [*5647, 5656, 6668, 8308, 8332*] and [*no other standard*].

FCS_SSH_EXT.1.2 The TSF shall ensure that the SSH protocol implementation supports the following authentication methods: [

- *"password" (RFC 4252),*
- *"publickey" (RFC 4252): [*
  - *rsa-sha2-256 (RFC 8332),*

  - *rsa-sha2-512 (RFC 8332),*
  - *ecdsa-sha2-nistp256 (RFC 5656),*
  - *ecdsa-sha2-nistp384 (RFC 5656),*

]

] and no other methods.

FCS_SSH_EXT.1.3 The TSF shall ensure that, as described in RFC 4253, packets greater than [*65,824 bytes*] in an SSH transport connection are dropped.

FCS_SSH_EXT.1.4 The TSF shall protect data in transit from unauthorised disclosure using the following mechanisms: [

- *aes128-cbc (RFC 4253),*
- *aes256-cbc (RFC 4253),*
- *aes256-gcm@openssh.com (RFC 5647)*

] and no other mechanisms.

FCS_SSH_EXT.1.5 The TSF shall protect data in transit from modification, deletion, and insertion using: [

- *hmac-sha2-256 (RFC 6668),*
- *hmac-sha2-512 (RFC 6668),*
- *implicit*

] and no other mechanisms.

FCS_SSH_EXT.1.6 The TSF shall establish a shared secret with its peer using: [

- *ecdh-sha2-nistp256 (RFC 5656),*
- *ecdh-sha2-nistp384 (RFC 5656),*
- *ecdh-sha2-nistp521 (RFC 5656),*

] and no other mechanisms.

FCS_SSH_EXT.1.7 The TSF shall use *SSH KDF* as defined in [

- *RFC 5656 (Section 4)*

] to derive the following cryptographic keys from a shared secret: *session keys*.

FCS_SSH_EXT.1.8 The TSF shall ensure that [

- *a rekey of the session keys,*

] occurs when any of the following thresholds are met:

- one hour connection time
- no more than one gigabyte of transmitted data, or
- no more than one gigabyte of received data.

5.3.2.19 FCS_SSHS_EXT.1 SSH Protocol - Server

FCS_SSHS_EXT.1.1 The TSF shall authenticate itself to its peer (SSH Client) using: [

- *rsa-sha2-256 (RFC 8332),*
- *rsa-sha2-512 (RFC 8332),*
- *ecdsa-sha2-nistp256 (RFC 5656),*
- *ecdsa-sha2-nistp384 (RFC 5656),*

].

5.3.2.20 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/IEC 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]* software based noise source] with a minimum of [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.

5.3.3 Identification and authentication (FIA)

5.3.3.1 FIA_AFL.1 Authentication Failure Handling

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* [*1*] and [*127*] characters.

5.3.3.3 FIA_PSK_EXT.1 (1): Pre-Shared Key Composition

FIA_PSK_EXT.1.1 (1) The TSF shall use PSKs for MKA as defined by IEEE 802.1X-2010, [[*IPsec protocols*]].

FIA_PSK_EXT.1.2 (1) The TSF shall be able to [*accept*] bit-based PSKs.

5.3.3.4 FIA_PSK_EXT.1 (2): Pre-Shared Key Composition

FIA_PSK_EXT.1.1 (2) The TSF shall be able to use pre-shared keys for IPsec and [*IKEv2*].

FIA_PSK_EXT.1.2 (2) The TSF shall be able to accept the following as pre-shared keys: [*generated bit-based*] keys.

5.3.3.5 FIA_ PSK_EXT.2: Generated Pre-Shared Keys

FIA_PSK_EXT.2.1 The TSF shall be able to [

- *accept externally generated pre-shared keys,*

]

5.3.3.6 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 actions on behalf of that administrative user.

FIA_UIA_EXT.1.3 The TSF shall provide the following remote authentication mechanisms [SSH password, SSH public key and no other mechanism] The TSF shall provide the following local authentication mechanisms [*password-based*]].

FIA_UIA_EXT.1.4 The TSF shall authenticate any administrative user's claimed identity according to each authentication mechanism specified in FIA_UIA_EXT.1.3.

5.3.3.7 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**.

5.3.3.8 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 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 [Certificate Revocation List (CRL) as specified in RFC 5759 Section 5].
- 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 DTLS/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.

5.3.3.9 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*].

5.3.3.10 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.

5.3.4 Security management (FMT)

5.3.4.1 FMT_MOF.1/ManualUpdate Management of Security Functions Behaviour

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

5.3.4.2 FMT_MOF. 1/Services Management of Security Functions Behaviour

FMT_MOF.1/Services The TSF shall restrict the ability to **start and stop** ~~the functions~~ **services** to *Security Administrators*.

5.3.4.3 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, handling of audit data] to *Security Administrators*.

5.3.4.4 FMT_MTD.1/CoreData Management of TSF Data

FMT_MTD.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 remotely;*
- *Ability to configure the access banner;*
- *Ability to configure the remote session inactivity time before session termination;*
- *Ability to update the TOE, and to verify the updates using digital signature capability prior to installing those updates;*
  [
  - Ability to start and stop services;
  - Ability to configure local audit behavior (e.g. changes to storage locations for audit; changes to behaviour when local audit storage space is full; changes to local audit storage size);
  - Ability to modify the behaviour of the transmission of audit data to an external IT entity;
  - Ability to manage the cryptographic keys;
  - Ability to configure the cryptographic functionality;
  - Ability to configure thresholds for SSH rekeying;
  - Ability to configure the lifetime for IPsec SAs;
  - Ability to re-enable an Administrator account;
  - Ability to configure NTP;
  - 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 generate Certificate Signing Request (CSR) and process CA certificate response;
  - Ability to administer the TOE locally;
  - Ability to configure the local session inactivity time before session termination or locking;
  - Ability to configure the authentication failure parameters for FIA_AFL.1;
  - Ability to manage the trusted public keys database;

  ].

5.3.4.7 FMT_SMF.1/MACSEC Specification of Management Functions (MACsec)

FMT_SMF.1.1/MACSEC The TSF shall be capable of performing the following management functions **related to MACsec functionality**: [*Ability of a Security Administrator to:*

- *Manage a PSK-based CAK and install it in the device*
- *Manage the key server to create, delete, and activate MKA participants* [*as specified in IEEE 802.1X-2020, Sections 9.13 and 9.16 (cf. MIB object ieee8021XKayMkaParticipantEntry) and section. 12.2 (cf. function createMKA*())]
- *Specify a lifetime of a CAK*
- *Enable, disable, or delete a PSK-based CAK using* [[*CLI management commands*]]
  [
- *No other MACsec management functions*
  ]].

5.3.4.8 FMT_SMF.1/VPN Specification of Management Functions

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]].*

5.3.4.9 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 remotely*

are satisfied.

5.3.5 Packet Filtering (FPF)

5.3.5.1 FPF_RUL_EXT.1 Packet Filtering Rules

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 protocols fields: [

- *IPv4 (RFC 791)*
  - *source address*
  - *destination address*
  - *protocol*
- *IPv6 (RFC 8200)*
  - *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_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.

5.3.6.2 FPT_FLS.1 Failure with Preservation of Secure State

FPT_FLS.1.1 The TSF shall **fail-secure** when **any of** 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].*

5.3.6.3 FPT_FLS.1/SelfTest Failure with Preservation of Secure State (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].*

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

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

5.3.6.5 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 [synchronise time with an NTP server].

5.3.6.6 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) to verify the integrity of the TOE firmware and software;
- Prior to providing any cryptographic service and [on-demand, [*periodically during normal operation*]] to verify correct operation of cryptographic implementation necessary to fulfil the TSF;
- [no other] self-tests [*none*]

to demonstrate the correct operation of the TSF: noise source health tests.

FPT_TST_EXT.1.2 The TSF shall respond to [all failures, [*AES KAT,* AES-GCM KAT, *RSA Signature KAT, RNG/DRBG KAT, HMAC KAT, SHA-1/256/384/512 KAT, ECDSA self-test, Software Integrity Test*]] by [rebooting].

5.3.6.7 FPT_TST_EXT.3: 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***].

5.3.6.8 FPT_TUD_EXT.1 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** [***no other mechanisms***] prior to installing those updates.

5.3.6.9 FPT_RPL.1 Replay Detection

FPT_RPL.1.1 The TSF shall detect replay for the following entities: [*MPDUs, MKA frames*].

FPT_ RPL.1.2 The TSF shall perform [*discarding of the replayed data, logging of the detected replay attempt*] when replay is detected.

5.3.6.10 FPT_RPL_EXT.1 Replay Protection for XPN

FPT_RPL_EXT.1.1 The TSF shall support extended packet numbering (XPN) as per IEEE 802.1AE-2018.

FPT_RPL_EXT.1.2 The TSF shall support [GCM-AES-XPN-128, GCM-AES-XPN-256] as per IEEE 802.1AE-2018.

5.3.6.11 FPT_CAK_EXT.1 Protection of CAK Data

FPT_CAK_EXT.1.1 The TSF shall prevent reading of CAK values by administrators.

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, [

- *terminate the session*]

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 ~~user~~ **Administrator**-initiated termination of the ~~user's~~ **Administrator**'s own interactive session.

5.3.7.4 FTA_TAB.1 Default TOE Access Banners

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

5.3.8 Trusted Path/Channels (FTP)

5.3.8.1 FTP_ITC.1 Inter-TSF trusted channel

FTP_ITC.1.1: The TSF shall **be capable of using** [***IPSEC***] **to** provide a **trusted** communication channel between itself and ~~another trusted IT product~~ **authorized IT entities supporting the following capabilities: audit server, [*authentication server [remote VPN Gateways, NTP Server]*]** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from ~~modification or~~ disclosure **and detection of modification of the channel data**.

FTP_ITC.1.2 The TSF shall permit [*the TSF,* ***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,*
- *NTP server using IPSec*

].

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

FTP_ITC.1.1/MACSEC The TSF shall provide a communication channel between itself and **a MACsec peer** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.

FTP_ITC.1.2/MACSEC The TSF shall permit [*the TSF, another trusted IT product*] to initiate communication via the trusted channel.

FTP_ITC.1.3/MACSEC The TSF shall initiate communication via the trusted channel for *[communication with MACsec peers that require the use of MACsec].*

5.3.8.3 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.1.2/VPN 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 or peers].*

5.3.8.4 FTP_TRP.1/Admin Trusted Path

FTP_TRP.1.1/Admin The TSF shall **be capable of using [IPsec, SSH] to** provide a trusted communication channel between itself and **authorized** remote **Administrators** ~~**users**~~ that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from disclosure **and provides detection of modification of the channel data**.

FTP_TRP.1.2/Admin The TSF shall permit remote **Administrators** ~~users~~ 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 v3.0e, PKG_SSH_V1.0, MOD_VPNGW v1.3 and MOD_MACSEC_V1.0 contain all the requirements claimed in this Security Target. As such the dependencies are not applicable since the PP and PP Modules have been approved.

5.5 Security Assurance Requirements

5.5.3 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 15 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 procedures
Life cycle support (ALC)	ALC_CMC.1	Labeling of the TOE
	ALC_CMS.1	TOE CM coverage
	ALC_FLR.2	Flaw reporting procedures
Tests (ATE)	ATE_IND.1	Independent testing - conformance
Vulnerability assessment (AVA)	AVA_VAN.1	Vulnerability survey

5.5.4 Security Assurance Requirements Rationale

The Security Assurance Requirements (SARs) in this Security Target represent the SARs identified in the NDcPPv3.0e, PKG_SSH_V1.0, MOD_VPNGW v1.3 and MOD_MACSEC_V1. 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 16 Assurance Measures

Component	How requirement will be met
ADV_FSP.1	There are no specific assurance activities associated with ADV_FSP.1. The requirements on the content of the functional specification information are implicitly assessed by virtue of the other assurance activities being performed. The functional specification is comprised of the information contained in the AGD_OPE and AGD_PRE documentation, coupled with the information provided in the TSS of the ST. The assurance activities in the functional requirements point to evidence that should exist in the documentation and TSS section; since these are directly associated with the SFRs, the tracing in element ADV_FSP.1.2D is implicitly already done and no additional documentation is necessary.
AGD_OPE.1	The Administrative Guide provides the descriptions of the processes and procedures of how the administrative users of the TOE can securely administer the TOE using the interfaces that provide the features and functions detailed in the guidance.
AGD_PRE.1	The Installation Guide describes the installation, generation, and startup procedures so that the users of the TOE can put the components of the TOE in the evaluated configuration.
ALC_CMC.1	The Configuration Management (CM) document(s) describes how the consumer (end-user) of the TOE can identify the evaluated TOE (Target of Evaluation). The CM document(s), identifies the configuration items, how those configuration items are uniquely identified, and the adequacy of the procedures that are used to control and track changes that are made to the TOE. This includes details on what changes are tracked, how potential changes are incorporated, and the degree to which automation is used to reduce the scope for error. The TOE will also be provided along with the appropriate administrative guidance.
ALC_CMS.1	
ALC_FLR.2	Cisco documents the flaw remediation and reporting procedures so that security flaw reports from TOE users can be appropriately acted upon, and TOE users can understand how to submit security flaw reports to the developer.
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 17 How TOE SFRs Measures

TOE SFRs	How the SFR is Met
FAU_GEN.1 FAU_GEN.1/MACSEC FAU_GEN.1/VPN	The TOE generates an audit record whenever any of the audited events in Table 14 occurs. The types of events that cause audit records to be generated include: startup and shutdown of the audit mechanism, cryptography related events, 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 and a key reference. Additionally, the startup and shutdown of the audit functionality is audited. The audit trail consists of the individual audit records; one audit record for each event that occurred. The audit record can contain up to 80 characters and a percent sign (%), which follows the time-stamp information. As noted above, the information includes at least all of the required information. Example audit events are included below: Nov 19 2022 13:55:59: %CRYPTO-6-SELF_TEST_RESULT: Self test info: (Self test activated by user: lab) Nov 19 2022 13:55:59: %CRYPTO-6-SELF_TEST_RESULT: Self test info: (Software checksum ... passed) Nov 19 2022 13:55:59: %CRYPTO-6-SELF_TEST_RESULT: Self test info: (DES encryption/decryption ... passed) Nov 19 2022 13:55:59: %CRYPTO-6-SELF_TEST_RESULT: Self test info: (3DES encryption/decryption ... passed) Nov 19 2022 13:55:59: %CRYPTO-6-SELF_TEST_RESULT: Self test info: (SHA hashing ... passed) Nov 19 2022 13:55:59: %CRYPTO-6-SELF_TEST_RESULT: Self test info: (AES encryption/decryption ... passed) In the above log events a date and timestamp is displayed as well as an event description "CRYPTO-6-SELF_TEST_RESULT: Self test info: (Self test)". The subject identity where a command is directly run by a user is displayed "user: lab." The outcome of the command is displayed: "passed". The logging buffer size can be configured from a range of 8192 (default) to 2147483647 bytes. 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. 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). 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. Refer to the Common Criteria Operational User Guidance and Preparative Procedures for command description and usage information.

TOE SFRs	How the SFR is Met
	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. Once the box is up and operational and the crypto self test command is entered, then the result messages would be displayed on the console and will also be logged. If the TOE encounters a failure to invoke any one of the cryptographic functions, a log record is generated.
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: Jun 18 2022 11:17:20.769: AAA/BIND(0000004B): Bind i/f Jun 18 2022 11:17:20.769: AAA/AUTHEN/LOGIN (0000004B): Pick method list 'default' Jun 18 2022 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 FAU_STG.1	The TOE is a standalone TOE and is capable of storing audit data locally. 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 uses a volatile log buffer to temporarily store audit records until communication with the syslog server is restored. When the connection is restored, the TOE will transmit the buffer contents when connected to the syslog server. For audit records stored internally to the TOE, the TOE implements a circular log file in persistent storage. When the configured maximum log size is reached, the oldest audit records are overwritten. The log file size is configurable by the administrator, with supported values ranging from: • Non-persistent: 4096 bytes (default) to 2147483647 • Persistent: 8192 bytes (default) to 2147483647 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	The TOE implements Diffie-Hellman (DH) groups 14, 15, 16, 19, and 20 key establishment schemes in accordance with NIST Special Publication 800-56A Revision 3 and RFC 3526. The TOE can operate as both a sender and a receiver for Diffie-Hellman-based key establishment schemes. The TOE complies with section 5.6 and all applicable subsections in NIST SP 800-56A concerning asymmetric key pair generation and key establishment. Asymmetric cryptographic keys used for IKE peer authentication are generated in accordance with FIPS PUB 186-5: RSA schemes follow Appendix B.3, and ECDSA schemes follow Appendix B.4. For key establishment schemes using safe-prime groups, the TOE adheres to the requirements in NIST SP 800-56A Revision 3. The TOE can generate: • RSA key pairs with key sizes of 2048 or 3072 bits, and • ECDSA key pairs using NIST curves P-256 and P-384.
FCS_CKM.1/IKE	
FCS_CKM.2	

<table>
<tr><th>TOE SFRs</th><th>How the SFR is Met</th></tr>
<tr><td></td><td>These key pairs serve multiple purposes:
• Generation of Certificate Signing Requests (CSRs), which can be sent to a Certificate Authority (CA) via the Simple Certificate Enrollment Protocol (SCEP),
• SSH host key authentication, where RSA and ECDSA key pairs are used by the TOE to authenticate itself during SSH connections,
• IKE peer authentication, and
• Digital signatures, using ECDSA and RSA algorithms compliant with FIPS PUB 186-4.

The TOE ensures the integrity of CSRs and received certificates using digital signatures, which protect the hash of the TOE's public key in transit. The TOE stores and manages the received X.509v3 certificates and can distribute them to external entities such as Registration Authorities (RAs).

Additionally, the TOE can use the X.509v3 certificates for securing IPsec sessions. The embedded PKI client functionality in IOS-XE provides secure mechanisms for certificate management, including distribution and revocation.

<table>
<tr><th>Scheme</th><th>SFR</th><th>Service</th></tr>
<tr><td rowspan="2">ECC</td><td>FCS_SSH_EXT.1</td><td>SSH Remote Administration</td></tr>
<tr><td>FCS_IPSEC_EXT.1</td><td>Trusted communications between the TOE and a(n):
• Audit server
• Authentication server
• Remote VPN Gateway
• NTP Server</td></tr>
<tr><td>FFC</td><td>FCS_IPSEC_EXT.1</td><td>Trusted communications between the TOE and a(n):
• Audit server
• Authentication server
• Remote VPN Gateway
• NTP Server</td></tr>
</table>
</td></tr>
<tr><td>FCS_CKM.4</td><td>The TOE meets all requirements specified in FIPS 140-2 for destruction of keys and Critical Security Parameters (CSPs). Refer to Table 18 for more information on the key zeroization.</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 5 for validation details. AES is implemented in the IPSec and SSH protocols. The TOE provides AES encryption and decryption in support of IPSec and SSHv2 for secure communications.</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 3072 as specified in ISO/IEC 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 and 384 bits as specified in FIPS PUB 186-4, "Digital Signature Standard". The TOE provides cryptographic signature services using ECDSA that meets ISO/IEC 14888-3, Section 6.4 with NIST curves P-256 and P-384 and P-521.

Please refer to Table 5 for all the CAVP references.</td></tr>
</table>

TOE SFRs	How the SFR is Met
FCS_COP.1/Hash FCS_COP.1/KeyedHash	The TOE provides cryptographic hashing services using SHA-1, SHA-256, SHA-384, and SHA-512 as specified in ISO/IEC 10118-3:2004. The TOE also provides keyed-hash message authentication services using HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512, which operate on 512-bit and 1024-bit blocks. The key sizes and message digest sizes are 160, 256, 384, and 512 bits respectively, as specified in ISO/IEC 9797-2:2011, Section 7 “MAC Algorithm 2”. Hash functions and keyed-hash algorithms are used throughout the TOE for the following cryptographic services: ● **IKE (ISAKMP) authentication**: Administrators may configure the TOE to use any of the HMAC-SHA variants (HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, or HMAC-SHA-512) for integrity protection of ISAKMP exchanges. ● **IPsec SA authentication**: The TOE supports esp-sha-hmac (HMAC-SHA-1), esp-sha256-hmac, and esp-sha512-hmac options for IPsec Security Association integrity protection. ● **SSH protocol support**: The TOE provides Secure Hash Standard (SHS) hashing in support of SSH for integrity and secure communication. ● **Digital signature verification**: The TOE uses SHA-based hash functions (SHA-256, SHA-384, and SHA-512) to verify RSA and ECDSA digital signatures, including: ○ **Trusted update mechanisms**, where the TOE verifies the integrity and authenticity of digitally signed software updates before installation. ○ **Software integrity self-tests**, performed at startup to verify the integrity of the running image using a digitally signed hash. ● **Cryptographic Known Answer Tests (KATs)**: The TOE performs SHA-based hashing as part of its self-tests during startup to validate the correctness of the cryptographic algorithm implementations. ● **RADIUS Key Wrap**: The TOE uses HMAC-SHA1 as part of the RADIUS key wrapping function. Management of cryptographic algorithms, including hash algorithm selection and configuration, is handled via the TOE’s CLI and is auditable. Please see CAVP certificates in Table 5 for validation details.
FCS_COP.1/CMAC FCS_COP.1/MACSEC	The TOE implements AES-CMAC keyed hash function for message authentication as described in NIST SP 800-38B. The key length, hash function used, block size, and output MAC length used are as follows: AES-128 (hash function and key length) Block Sizes: 128 bits (block size) Message Length: 128 bits (output MAC length) AES-256 (hash function and key length) Block Sizes: 128 bits (block size) Message Length: 128 bits (output MAC length) The TOE provides symmetric encryption and decryption capabilities using AES in AES Key Wrap and GCM mode (128 and 256 bits) as described in AES as specified in ISO 18033-3, AES Key Wrap as specified in NIST SP 800-38F, GCM as specified in ISO 19772. AES is implemented in MACsec protocol. The relevant FIPS certificate numbers are listed in Table 6 FIPS References.

TOE SFRs	How the SFR is Met
FCS_RBG_EXT.1	The TOE implements a NIST-approved AES-CTR Deterministic Random Bit Generator (DRBG), as specified in ISO/IEC 18031:2011 seeded by an entropy source that accumulates entropy from a software-based noise source (for CSfC purposes, AES-256). 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 as specified in RFC 4301. The IPsec implementation provides both VPN peer-to-peer TOE capabilities. The VPN peer-to-peer tunnel allows 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. In addition to tunnel mode, which is the default IPsec mode, the TOE also supports transport mode, allowing for only the payload of the packet to be encrypted. If tunnel mode is explicitly specified, the router will request tunnel mode and will accept only tunnel mode. The TOE implements IPsec to provide both certificates and pre-shared key-based authentication and encryption services to prevent unauthorized viewing or modification of data as it travels over the external network. The TOE implementation of the IPsec standard (in accordance with the RFCs noted in the SFR) uses the Encapsulating Security Payload (ESP) protocol to provide authentication, encryption and anti-replay services. The IPsec protocol ESP, as defined by RFC 4303, is implemented 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 (RFC 3602), AES-GCM-192 (RFC 4106) together with a Secure Hash Algorithm (SHA)-based HMAC HMAC-SHA1, HMAC-SHA-256, HMAC-SHA-512. 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. When certificates are used for authentication, the TOE matches the full Distinguished Name (DN) in the certificate against a configured reference identifier. The certificate is considered valid if the DN naming attributes match exactly in both type and value. Although the DN may include a Common Name (CN) field — such as a CN containing an FQDN, user FQDN, or IP address — the TOE performs matching only on the complete DN. It does not extract or compare CN values independently. IKE separates negotiation into two phases: phase 1 and phase 2. Phase 1 creates the first tunnel, which protects later ISAKMP negotiation messages. The key negotiated in phase 1 enables IKE peers to communicate securely in phase 2. During Phase 2 IKE establishes the IPsec SA. IKE maintains a trusted channel, referred to as a Security Association (SA), between IPsec peers that is also used to manage IPsec connections, including: • The negotiation of mutually acceptable IPsec options between peers (including peer authentication parameters, either signature based or pre-shared key based), • The establishment of additional Security Associations to protect packets flows using Encapsulating Security Payload (ESP), and • The agreement of secure bulk data encryption AES keys for use with ESP. After the two peers agree upon a policy, the security parameters of the policy are identified by an SA established at each peer, and these IKE SAs apply to all subsequent IKE traffic during the negotiation.

TOE SFRs	How the SFR is Met
	The TOE supports IKEv2 session establishment and allows configuration of lifetimes for both Phase 1 SAs and Phase 2 SAs using the lifetime command. Phase 1 SA lifetimes can be set for up to 24 hours, and Phase 2 SA lifetimes can be set for up to 8 hours. Phase 2 SA lifetimes can also be configured by an Administrator based on the number of bytes. The TOE supports Diffie-Hellman Groups 14, 15, 16, 19, and 20. Group 14 (2048-bit keys) can be set by using the “group 14” command in the config mode. The nonces used in IKE exchanges are generated in a manner such that the probability that a specific nonce value will be repeated during the life of a specific IPsec SA is less than 1 in 2^[112]. Nonce lengths are dynamically determined based on the negotiated Diffie-Hellman group or the pseudorandom function (PRF) hash as follows: • Nonce lengths must be sufficiently large to support all TOE-chosen proposals during the exchange, as nonces may be exchanged before the Diffie-Hellman group is negotiated. • For Diffie-Hellman Group 20, which has a security strength of 192 bits, the nonce length is at least 192 bits. • For Diffie-Hellman Groups 14, 15, 16, and 19, the nonce lengths are determined based on the security strengths associated with their respective group sizes. • When the negotiated PRF hash is used to determine nonce length, the nonce is set to at least half the output size of the PRF hash: ○ For HMAC-SHA-384, the nonce length is at least 192 bits. ○ For HMAC-SHA-512, the nonce length is at least 256 bits. 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 Deterministic Random Bit Generator (DRBG). The TOE supports configuring the maximum amount of traffic that is allowed to flow for a given IPsec (Phase 2) 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, and AES-GCM-256 for encrypting IKEv2 payloads. The administrator is instructed in the AGD to ensure that the size of the key used for ESP is greater than or equal to the key size used to protect the IKEv2 payload. The TOE supports Diffie-Hellman Group 14 (2048-bit keys), 19 (256-bit Random ECP), 20 (384-bit Random ECP), 15 (3072-bit MODP), and 16 (4096-bit MODP) in support of IKE Key Establishment. These keys are generated using the AES-CTR Deterministic Random Bit Generator (DRBG), as specified in SP 800-90, 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), 384 (for DH Group 20), 256 (for DH Group 15) and 256 (for DH Group 16) bits. The DH group can be configured by issuing the following command during the configuration of IPsec: Router (config)# group 14 This command selects DH Group 14 (2048-bit MODP) for IKE and this sets the DH group offered during negotiations. 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

TOE SFRs	How the SFR is Met
	specify the keying material to be used. SAs are unidirectional and are established per security protocol (ESP). Privileged administrators can define the traffic that needs to be protected between two IPsec peers by configuring a crypto map, access lists, and then applying the access lists to the crypto map. The access list 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 The match address 101 command means to use access list 101 in order to determine which traffic is relevant. For example: Router (config-crypto-map)#match address 101 The traffic matching the permit acls would then flow through the IPSec tunnel and be classified as “PROTECTED”. Traffic that does not match a permit acl and is also blocked by other non-crypto acls on the interface would be DISCARDED. 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 entry was not configured that matches the ping traffic. 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. 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. 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. RFC 8784 (Mixing Pre-shared Keys in IKEv2 for Postquantum Security) describes an extension to the IKEv2 protocol to allow it to be resistant to a quantum computer by using pre-shared keys known as PPKs. The RFC defines negotiation of PPK capability, communication of PPK ID, mixing of PPK as an additional input in the session key derivation, and optional fallback to non-PPK-based session.
FCS_MACSEC_EXT.1	The TOE implements MACsec in compliance with IEEE Standard 802.1AE-2018. The MACsec connections maintain confidentiality of transmitted data and to take measures against frames

TOE SFRs	How the SFR is Met
	transmitted or modified by unauthorized devices. The SCI is composed of a globally unique 48-bit MAC Address and the Secure System Address (port). The SCI is part of the SecTAG if the SC bit is set and will be at the end of the tag. Any MPDUs during a given session that contain an SCI other than the one used to establish that session are rejected. Only EAPOL (PAE EtherType 88-8E), MACsec frames (EtherType 88-E5), MAC control frames EtherType 0x876F are permitted and others are rejected. The EtherType 0x876F has been officially allocated to Cisco for WAN MACsec applications.
FCS_MACSEC_EXT.2	The TOE implements the MACsec requirement for integrity protection with the confidentiality offsets of 0, 30 and 50 through the CLI command of “mka-policy confidentiality-offset command”. An offset value of 0 does not offset the encryption and offset values of 30 and 50 offset the encryption by 30 and 50 characters respectively. An Integrity Check Value (ICV) that is 16 bytes in length is derived with the Secure Association Key (SAK) and is used to provide assurance of the integrity of MPDUs. The TOE derives the ICV from a CAK using KDF, using the SCI as the most significant bits of the IV and the 32 least significant bits of the PN as the IV.
FCS_MACSEC_EXT.3	Each SAK is generated using the KDF specified in IEEE 802.1X-2010 section 6.2.1 using the following transform - KS-nonce = a nonce of the same size as the required SAK, obtained from an RNG each time an SAK is generated. The TOE’s random bit generator is used for creating these unique nonces. Each of the keys used by MKA is derived from the CAK. The key string is the CAK that is used for ICV validation by the MKA protocol. The CAK is not used directly but derives two further keys from the CAK using the AES cipher in CMAC mode. The derived keys are tied to the identity of the CAK and thus restricted to use with that particular CAK. These are the ICV Key (ICK) used to verify the integrity of MPDUs and to prove that the transmitter of the MKPDU possesses the CAK, and the Key Encrypting Key (KEK) used by the Key Server, elected by MKA, to transport a succession of SAKs, for use by MACsec, to the other member(s) of a CA. The size of the key is based on the configured AES key sized used. If using AES 128-bit CMAC mode encryption, the key string will be 32-bit hexadecimal in length. If using 256-bit encryption, the key string will be 64-bit hexadecimal in length.
FCS_MACSEC_EXT.4	MACsec peer authentication is achieved by only using pre-shared keys. The SAKs are distributed between these peers using AES Key Wrap. Prior to distribution of the SAKs between these peers, the TOE uses AES Key Wrap in accordance with AES as specified in ISO 18033-3, AES in CMAC mode as specified in NIST SP 800-38B, and GCM as specified in ISO 19772. The “Key-chain macsec lifetime” key configuration command is used to specify the lifetime for CAKs. The “MACSEC Key-chain key” configuration command is used to specify the length of the CKN that is allowed to be between 1 and 32 octets.
FCS_MKA_EXT.1	The TOE implements Key Agreement Protocol (MKA) in accordance with IEEE 802.1X-2010 and 802.1Xbx-2014. The TOE enforces an MKA Lifetime Timeout limit of 6.0 seconds and a Hello Timeout limit of 2.0 seconds, as part of the MACsec Key Agreement (MKA) protocol. These timeouts ensure timely

TOE SFRs	How the SFR is Met
	revalidation of Secure Association Keys (SAKs) and control protocol behavior in accordance with IEEE 802.1X-2010 and 802.1Xbx-2014. The TOE discards MKPDUs that do not satisfy the requirements listed under FCS_MKA_EXT.1.7 in Section 5.3.2.16. All valid MKPDUs that meet the requirements as defined under FCS_MKA_EXT.1.7 are decoded in a manner conformant to IEEE 802.1x-2010 Section 11.11.4. On successful peer authentication, a connectivity association is formed between the peers and a secure Connectivity Association Key Name (CKN) is exchanged. After the exchange, the MKA ICV is validated with a Connectivity Association Key (CAK), which is effectively a secret key. For the Data Integrity Check, MACsec uses MKA to generate an Integrity Check Value (ICV) for the frame arriving on the port. If the generated ICV is the same as the ICV in the frame, then the frame is accepted; otherwise, it is dropped. The key string is the Connectivity Association Key (CAK) that is used for ICV validation by the MKA protocol.
FCS_SSH_EXT.1 FCS_SSHS_EXT.1	The TOE implementation of SSHv2 complies with RFCs 4251, 4252, 4253, 4254, 5647, 5656, 6668, 8308 section 3.1, 8332 and supports the following: • The TOE's implementation of SSH public-key based authentication supports `rsa-sha2-256`, `rsa-sha2-512`, `ecdsa-sha2-nistp256`, and `ecdsa-sha2-nistp384`. • 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. • Local password-based authentication for administrative users accessing the TOE through SSHv2 and optionally supports deferring authentication to a remote AAA server. • Encryption algorithms, `aes128-cbc`, `aes256-cbc`, and `aes256-gcm@openssh.com` to ensure confidentiality of the session. (Note: Changed from AES-CBC-128 and AES-CBC-256 to the RFC-compliant naming convention.) • The TOE's implementation of SSHv2 supports hashing algorithms `hmac-sha2-256` and `hmac-sha2-512` to ensure the integrity of the session. • The TOE's implementation of SSHv2 can be configured to allow `ecdh-sha2-nistp256`, `ecdh-sha2-nistp384`, and `ecdh-sha2-nistp521` Key Establishment. • The TOE supports Key Derivation Functions (KDFs) as specified in RFC 4253, which includes the use of KDFs derived from the specified RFCs to ensure secure key establishment. • Packets greater than 65,824 bytes in an SSH transport connection are dropped. Large packets are detected by the SSH implementation and dropped internally to the SSH process. • The TOE can also be configured to ensure that SSH re-keying occurs no longer than once per hour, no more than one gigabyte of transmitted data, and no more than one gigabyte of received data for the session key. Rekeying is performed upon reaching the threshold that is hit first. Please refer to Table 5 for all the CAVP references.
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. All successive unsuccessful authentication attempts are logged on the router. When a privileged administrator or non-privileged administrator attempting to log into the administrative CLI reaches the administratively set maximum number of failed authentication

TOE SFRs	How the SFR is Met
	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. 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 supports passwords of 1 to 127 characters.
FIA_PSK_EXT.1 (1) FIA_PSK_EXT.1 (2) FIA_PSK_EXT.2	Through the implementation of the CLI, the TOE supports mixing pre-shared keys in IKEv2 for Postquantum Security (RFC 8784) and for authentication of IPsec tunnels. The TOE accepts bit-based pre-shared keys entered exclusively as fixed-length hexadecimal strings. The TOE supports 64-character hexadecimal PSKs (representing 256-bit keys) for IPsec. The data that is input is conditioned by the cryptographic module prior to use via SHA-1. The TOE supports use of pre-shared keys for MACsec key agreement protocols as defined by IEEE 802.1X. The pre-shared keys are not generated by the TOE, but the TOE accepts the keys in the form of HEX strings. This is done via the CLI configuration command ‘key chain test_key macsec’. The TOE accepts pre-shared keys that are 32 characters in length for AES 128-bit CMAC mode encryption and pre-shared keys that are 64 characters in length for AES 256-bit CMAC mode encryption.
FIA_UIA_EXT.1	The TOE requires all users to be successfully identified and authenticated before allowing any TSF-mediated actions to be performed, with the exception of the login warning banner that is displayed prior to user authentication. Administrative access to the TOE is facilitated through its Command Line Interface (CLI), which can be accessed either locally via a directly connected console or remotely via a secure SSHv2 session. The TOE mediates all administrative actions through the CLI. **Local Console Authentication** When using a directly connected console, the TOE prompts the administrator for a username and password. If the credentials match an entry in the local user database, access to the TOE is granted. A successful login occurs when the TOE verifies that the username exists in the local database and the provided password matches the stored credential. **Remote SSH Authentication** For remote SSH access, the TOE supports both username/password authentication and SSH public key authentication. • SSH password authentication follows the same process as the local console: the user provides a username and is prompted for a password. SSH public key authentication allows an administrator to authenticate by presenting a valid private key that corresponds to a previously configured public key in the TOE’s user database. A successful login occurs when the TOE verifies that the signature generated using the private key corresponds to the stored public key for the provided username. If the authentication mechanism fails (e.g., no key match or incorrect password), access is denied. The TOE does not provide any details about the reason for the failure. **RADIUS Authentication**

TOE SFRs	How the SFR is Met
	The TOE supports RADIUS-based remote authentication via an AAA server. When RADIUS is used, the TOE prompts the user for a username and password. These credentials are forwarded to the configured RADIUS server, which determines the outcome of the authentication. A successful login via RADIUS is determined by the TOE receiving an Access-Accept message from the RADIUS server in response to the submitted credentials. Only after receiving this response does the TOE grant administrative access. If an Access-Reject or no response is received, login is denied. The TOE does not process or evaluate the credentials locally in this case. Interfaces can be configured to attempt authentication through one or more remote RADIUS servers and fall back to local authentication only if the servers are unreachable (e.g., due to timeout or communication 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 FIA_X509_EXT.2 FIA_X509_EXT.3	The TOE uses X.509v3 certificates as defined by RFC 5280 (and RFC 8603 for CSfC purposes) to support authentication for IPsec connections. The TOE supports the following methods to obtain a certificate from a CA: • Simple Certificate Enrollment Protocol (SCEP)—A Cisco-developed enrollment protocol that uses HTTP to communicate with the CA or registration authority (RA). • Imports certificates in PKCS12 format from an external server • IOS-XE File System (IFS)—The router uses any file system that is supported by Cisco IOS-XE software (such as TFTP, FTP, flash, and NVRAM) to send a certificate request and to receive the issued certificate. • Manual cut-and-paste—The router 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 router and CA • Enrollment profiles—The router sends HTTP-based enrollment requests directly to the CA server instead of to the RA-mode certificate server (CS). • Self-signed certificate enrollment for a trust point All the certificates include at least the following information: public key, Common Name, Organization, Organizational Unit and Country. Public key infrastructure (PKI) credentials, such as Rivest, Shamir, and Adelman (RSA) and Elliptical Curve Digital Signature Algorithm (ECDSA) keys and certificates can be stored in a specific location on the TOE. Certificates are stored to NVRAM by default. 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 the hash value would then be invalid. 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. 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: • alt-subject-name • expires-on • issuer-name • name • serial-number • subject-name • unstructured-subject-name • valid-start 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. The physical security of the TOE (A.PHYSICAL_PROTECTION) protects the router 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. CRL is configurable and may be used for certificate revocation. The authorized administrator could use the "revocation-check" command to specify at least one method of revocation checking; CRL is the default method and must be selected in the evaluated configuration as the 'none' option is not allowed. The authorized administer sets the trust point and its name and the revocation-check method. The extendedKeyUsage field is validated according to the following rules: • Certificates used for trusted updates and executable code integrity verification have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) • Server certificates 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 have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) • OCSP certificates presented for OCSP responses have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field. 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. 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. During IPsec connection establishment using certificate-based authentication, the TOE performs X.509 certificate path validation as specified in RFC 5280. As part of this process, the TOE checks the revocation status of each certificate in the chain—starting from the peer's certificate up to the trust anchor—using Certificate Revocation Lists (CRLs). This revocation checking occurs at the time

TOE SFRs	How the SFR is Met
	the certificate is used (i.e., during the IKE negotiation), and the connection will be rejected if any certificate in the chain is found to be revoked or if the CRL cannot be retrieved. The revocation-check command is used to configure this behavior and must be set to crl in the evaluated configuration. The TOE performs validation of the basicConstraints extension and CA flag for subordinate CA certificates during the crypto pki authenticate process. However, for self-signed root CA certificates, this validation is not performed automatically by the TOE at the crypto pki authenticate stage. In such cases, the administrator is required to manually verify the basicConstraints: CA:TRUE property of the self-signed root CA certificate to ensure compliance with FIA_X509_EXT.1.1/Rev and FIA_X509_EXT.1.2/Rev.
FMT_MOF.1/ManualUpdate	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. 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-XE 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. The Security Administrator has the ability to start and stop logging services on the TOE. Management functionality of the TOE is provided through the TOE CLI. 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. Therefore, semi-privileged administrators with only a subset of privileges can also modify TOE data based on if granted the privilege. No administrative functionality is available prior to administrative login. The Security Administrator is able to manage the cryptographic keys (generating keys, importing keys, or deleting keys) that are used in IPsec, MACSEC, and SSH communications. These keys can be managed via CLI as part of following operations: • SSH session keys– as part of session establishment and termination • SSH public/private keys – generate keypair, import/export public keys, public key-based authentication • IPsec keys – as part of IPsec session establishment and termination • Zeroize – delete keys The security administrator configures persistent IPsec keys using the 'crypto ikev2 keyring' command for IKEv2. Persistent MACsec keys are configured by the Security Administrator using the 'mka policy' and 'mka pre-shared-key' commands within a key chain for the MACsec Key Agreement (MKA) protocol. These configurations are stored in the router’s NVRAM, ensuring keys are retained across reloads. During operation, CAKs may also be loaded into internal ASIC registers for high-speed runtime use.
FMT_MOF.1/Services	
FMT_MOF.1/Functions	
FMT_MTD.1/CoreData	
FMT_MTD.1/CryptoKeys	
FMT_SMF.1	
FMT_SMF.1/MACSEC	

TOE SFRs	How the SFR is Met
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 - • The ability to administer the TOE remotely; • The ability to configure the access banner; • The ability to configure the remote session inactivity time before session termination; • The ability to update the TOE, and to verify the updates using digital signature capability prior to installing those updates; • The ability to start and stop services; • The ability to configure audit behavior (e.g. changes to storage locations for audit; changes to behavior when local audit storage space is full); • The ability to modify the behaviour of the transmission of audit data to an external IT entity; • The ability to manage the cryptographic keys; • The ability to configure the cryptographic functionality; • The ability to configure thresholds for SSH rekeying; • The ability to configure the lifetime for IPsec SAs; • The ability to re-enable an Administrator account; • The ability to configure NTP; • The ability to configure the reference identifier for the peer; • The ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors; • The ability to generate Certificate Signing Request (CSR) and process CA certificate response; • The ability to administer the TOE locally; • The ability to configure the local session inactivity time before session termination or locking; • The ability to configure the authentication failure parameters for FIA_AFL.1; • The ability to manage the trusted public keys database. Management functionality related to MACsec: • The ability to manage a PSK-based CAK and install it in the device; • The ability to manage the Key Server to create, delete, and activate MKA participants as specified in 802.1X, sections 9.13 and 9.16 (cf. MIB object ieee8021XKayMkaParticipantEntry) and section 12.2 (cf. function createMKA())]; • The ability to specify a lifetime of a CAK; • The ability to enable, disable, or delete a PSK-based CAK using a CLI management command. Management functionality related to VPN: • The ability to define packet filtering rules; • The ability to associate packet filtering rules to network interfaces; • The ability to order packet filtering rules by priority. 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-XE 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.

TOE SFRs	How the SFR is Met
	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. The privilege level determines the functions the user can perform; hence the Security Administrator with the appropriate privileges. The TOE can and shall be configured to authenticate all access to the command line interface using a username and password. The TOE supports both local administration via a directly connected console cable and remote administration via SSH.
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 the ‘ip access-group’ command. 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. 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. 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: 1. presumed address of source 2. presumed address of destination 3. transport layer protocol (or next header in IPv6) 4. Service used (UDP or TCP ports, both source and destination) 5. Network interface on which the connection request occurs These rules are supported for the following protocols: RFC 791(IPv4); RFC 8200 (IPv6); RFC 793 (TCP); RFC 768 (UDP). TOE compliance with these protocols is verified via regular quality assurance, regression, and interoperability testing. The following protocols are not supported and will be dropped before the packet is matched to an ACL; therefore, any “permit” or “deny” entries in an ACL will not show matches in the output of the ‘show ip access-list’ command. • IPv4 - Protocol 2 (IGMP) Protocol 2 is configuration dependent and is not supported when the device is not participating in an IGMP routing group. • IPv6 - 51 (AH), 60 (IPv6-Opts) Note: While IPv6 Protocols 43 (IPv6-Route), 44 (IPv6-Frag), and 135 (Mobility Header) are processed by the TOE when well-formed, they require specific keyword syntax in ACLs for proper filtering. The TOE's logging for these encapsulating protocols will reflect the inner encapsulated protocol, not the outer layer. For IPv6 Protocol 41 (Encapsulated IPv6-in-IPv6), the TOE's ACL logging will consistently display "0" as the protocol ID, regardless of the encapsulated protocol. However, filtering rules applied to Protocol 41 will function as expected.

TOE SFRs	How the SFR is Met
	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. 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. 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; 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; 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; 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; These interfaces reject requests in which the subject specifies the route for information to flow when it is in route to its destination. 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. 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. During initialization/startup (while the TOE is booting) the configuration has yet to be loaded, and no traffic can flow through any of its interfaces. No traffic can flow through the TOE interfaces until the POST has completed, and the configuration has been loaded. If any aspect of the POST fails during boot, the TOE will reload without forwarding traffic. If a critical component of the TOE, such as the clock or cryptographic modules, fails while the TOE is in an operational state, the TOE will reload, which stops the flow of traffic.
FPT_FLS.1	Whenever a failure occurs within the TOE that results in the TOE ceasing operation, the TOE securely disables its interfaces to prevent the unintentional flow of any 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.
FPT_SKP_EXT.1	

TOE SFRs	How the SFR is Met
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. 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 router obfuscates all clear-text passwords in the running or startup configuration—including line passwords, local-user and enable passwords, and all authentication key passwords—once you enable the feature globally with the command: 'service password-encryption' There are no administrative interfaces available that allow passwords to be viewed as they are encrypted via the password-encryption service. The router encrypts all supported secrets—IPsec preshared keys (PSKs), RADIUS/TACACS+ keys, local-user and enable passwords, line-level secrets using AES once you enable the feature globally with the commands: 'password encryption aes' This command activates the AES engine; you then set the AES master key (stored securely and not shown in the config) with the command: 'key config-key password-encrypt <your-master-key>' All existing and future secrets are stored in encrypted form and can only be decrypted on devices configured with the same master key. The TOE ensures that plaintext user passwords will not be disclosed even to administrators. To store enable and local-user passwords as non-reversible SHA256 hashes, configure each secret directly with the command: 'enable algorithm-type sha256 secret <plaintext>' 'username <user> algorithm-type sha256 secret <plaintext' These commands convert the specified passwords into salted, iterated SHA-256 hashes that cannot be decrypted. Additionally, enabling the 'hidekeys' command in the logging configuration ensures that and passwords are not displayed in plaintext.
FPT_STM_EXT.1	The TOE provides a source of date and time information that is used across various security functions. This time source is used to: • Timestamp TOE-generated audit records. • Track and enforce inactivity timeouts for administrative sessions. • Evaluate the validity period of X.509 certificates used in IPsec and SSH connections. • Enforce authentication and session timeouts, including AAA server response timeout, administrative session timeout, and SSH rekey intervals.
FCS_NTP_EXT.1	

TOE SFRs	How the SFR is Met
	- Perform cryptographic operations that rely on timing, such as IKE SA expiration and rekeying. - Provide time-based behavior within routing protocols (e.g., OSPF and BGP) where applicable. The TOE synchronizes with an NTP server for its reliable and accurate timestamp. The TOE can be configured to support at least three (3) NTP servers. The TOE supports NTPv4 and validates the integrity of the time-source using IPsec to provide trusted communication between itself and an NTP time source. In addition, the TOE does not allow the timestamp to be updated from broadcast addresses. NTP use UTC to synchronize computer clock times to a millisecond, and sometimes to a fraction of a millisecond.
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. Digital signatures 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. 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. Further instructions for how to do this verification are provided in the administrator guidance for this evaluation. Software images are available from Cisco.com at the following: http://www.cisco.com/cisco/software/navigator.html
FPT_TST_EXT.1 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. During the system bootup process (power on or reboot), all the Power on Startup Test (POST) components for the cryptographic module perform the POST for the corresponding component (hardware or software). These tests include: - Noise Source Health Tests – The Noise Source Health Tests check the functioning of the Noise Source that supplies randomness to the Entropy Source. The tests are designed to detect failure of the Noise Source. These tests are run at startup and continuously during normal operation. - AES Known Answer Test – 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

TOE SFRs	How the SFR is Met
	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. • AES-GCM Known Answer Test – In the encryption test, a known key, initialization vector (IV), and associated data (AAD) are used to encrypt a known plaintext, resulting in both a ciphertext and an authentication tag. These outputs are compared to known values to verify correct encryption and authentication. For the decryption test, the known key, IV, AAD, ciphertext, and tag are used to decrypt the ciphertext. The resulting plaintext is compared to the known plaintext to confirm correct decryption and authentication. • RSA Signature Known Answer Test (both signature/verification) – 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. • RNG/DRBG Known Answer Test – 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. • HMAC Known Answer Test – 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. • SHA-1/256/384/512 Known Answer Test – 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. • ECDSA self-test (both signature/verification) – 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. • Software Integrity Test – The Software Integrity Test is run automatically whenever the IOS system images is loaded and confirms that the image file that's about to be loaded has maintained its integrity. 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. If any component reports failure for the POST, the system crashes and appropriate information is displayed on the screen and saved in the crashinfo file. All ports are blocked from moving to forwarding state during the POST. If all components of all modules pass the POST, the system is placed in FIPS PASS state and ports are allowed to forward data traffic. If an error occurs during the self-test, a SELF_TEST_FAILURE system log is generated. Example Error: *Nov 26 2022 16:28:23.629: %CRYPTO-0-SELF_TEST_FAILURE: Encryption self-test failed

TOE SFRs	How the SFR is Met
	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.
FPT_RPL.1 FPT_RPL_EXT.1	MPDUs are replay protected in the TOE. Per IEEE 802.1AE-2018, each MAC Protocol Data Unit (MPDU) includes a Packet Number (PN) within the SecTag in its header for replay protection. The receiving device checks this PN against expected values and a replay window to detect and reject replayed frames. This mechanism ensures MPDUs are authenticated and protects the integrity of data transmission against replay attacks. The MKA frames are guarded against replay (If a MKPDU with duplicate MN (message number) and not latest MN comes along, then this MKPDU will be dropped and not processed further). MKA frames are checked for replay by checking the MN plus the MI for the participant. Replayed data is discarded and logged by the TOE. Extended Packet Numbering (XPN) uses a 64-bit Sequence Number (SN) field and an anti-replay window to detect replayed MPDUs. The device checks the received SN against the anti-replay window of the Inbound Secure Association (SA). If the SN falls outside the window, indicating a replayed frame, the MPDU is discarded. This mechanism ensures that only valid and non-replayed frames are processed, thereby securing the TSF against replay attacks.
FPT_CAK_EXT.1	A CAK value is specified in the configuration file by the Administrator using a bit-based (hex) format. The interface specifically implemented in the TSF for viewing the configuration file is the “show running-config” or “show startup-config “CLI commands. When the TOE is operating in the evaluated configuration, and the Administrator executes the “show running-config” or “show startup-config” CLI commands, the CAK data will not be displayed. This protects the CAK data from unauthorized disclosure.
FTA_SSL_EXT.1 FTA_SSL.3	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. 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_SSL.4	An administrator can 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.
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 FPT_ITC.1/MACSEC FTP_ITC.1/VPN	The TOE requires that peers and other TOE instances establish an IKE/IPSec connection in order to forward routing tables used by the TOE. The TOE protects communications between the TOE and the remote audit server using IPsec. This provides a secure channel to transmit the log events. Likewise, communications between the TOE and AAA servers are secured using IPsec. The TOE protects communications between itself and the NTP server using NTPv4, IPsec. The distinction between “remote VPN gateway” and “another instance of the TOE” is that “another instance of the TOE” would be installed in the evaluated configuration, and likely

TOE SFRs	How the SFR is Met
	administered by the same personnel, whereas a “remote VPN gateway/peer” could be any interoperable IPsec gateway/peer that is expected to be administered by personnel who are not administrators of the TOE, and who share necessary IPsec tunnel configuration and authentication credentials with the TOE administrators. For example, the exchange of X.509 certificates for certificate based authentication. MACsec is also used to secure communication channels between MACsec peers at Layer 2. The TOE acts as a server for both IPsec and MACsec secure channels.
FTP_TRP.1/Admin	All remote administrative communications take place over a secure encrypted SSHv2 session which has the ability to be encrypted further using IPsec. The SSHv2 session is encrypted using AES encryption. The remote users can initiate SSHv2 communications with the TOE.

7 Annex A: Key Zeroization

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

Table 18 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. 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. Overwritten with: 0x00
Skeyid	This is the IKE intermittent value used to create skeyid_d. This key is stored in SDRAM.	Automatically after IKE session terminated. Overwritten with: 0x00
skeyid_d	This is the IKE intermittent value used to derive keying data for IPsec. This key is stored in SDRAM.	Automatically after IKE session terminated. Overwritten with: 0x00
IKE session encrypt key	This is the IPsec key used for encrypting the traffic in an IPsec connection. This key is stored in SDRAM.	Automatically after IKE session terminated. Overwritten with: 0x00
IKE session authentication key	This is the IPsec key used for authenticating the traffic in an IPsec connection. This key is stored in SDRAM.	Automatically after IKE session terminated. Overwritten with: 0x00
IKEv2 Pre-Shared Key	The pre-shared key used for IKEv2 authentication. This key is stored in NVRAM.	Zeroized using the following command: **# no crypto ikev2 keyring <name> or by removing the key configuration** 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. 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. 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: **# crypto key zeroize ecdsa**[1] Overwritten with: 0x0d

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

Name	Description of Key	Zeroization
IKE RSA Private Key	The RSA private-public key pair is created by the device itself using the key generation CLI described below. 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. 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: **# crypto key zeroize rsa** 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. 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. 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: **# no radius-server key** Overwritten with: 0x0d
SSH Private Key	Used in establishing a secure SSH session This key is stored in NVRAM.	Zeroized using the following command: **# crypto key zeroize rsa** Overwritten with: 0x00
SSH Session Key	Used to encrypt SSH traffic This key is stored in SDRAM.	Overwritten automatically with 0x00 when the SSH trusted channel is no longer in use. Overwritten with: 0x00
MACsec SAK	The SAK is used to secure the control plane traffic. This key is stored in internal ASIC register.	Overwritten automatically when MACsec session expires.
MACsec CAK	The CAK secures the control plane traffic. Stored persistently in NVRAM as part of the configuration and also loaded into internal ASIC registers at runtime for processing efficiency.	Overwritten with a new value of the key. # key-string <32 or 64 hex-bit CAK>
MACsec Key Encryption Key (KEK)	The Key Encrypting Key (KEK) is used by Key Server, elected by MKA, to transport a succession of SAKs, for use by MACsec, to the other member(s) of a Secure Connectivity Association (CA). This key is stored in internal ASIC register.	Automatically when MACsec session expires.

Name	Description of Key	Zeroization
		The value is zeroized by overwritten by another key or freed upon session is expired.
MACsec Integrity Check Key (ICK)	The ICK is used to verify the integrity of MPDUs and to prove that the transmitter of the MKPDU possesses the CAK, this key is stored in internal ASIC register.	Automatically when MACsec session expires

8 Annex B: References

The following documentation was used to prepare this ST:

Table 19 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, Version 3.0e, 06 December 2023
[MOD_VPNGW]	PP-Module for Virtual Private Network (VPN) Gateways (MOD_VPNGW), Version 1.3, August 16, 2023
[MOD_MACSEC]	PP-Module for MACsec Ethernet Encryption Version 1.0 (MOD_MACSEC), Version 1.0, March 3, 2023
[PKG_SSH]	Functional Package for Secure Shell (SSH), Version 1.0, 13 May, 2021
[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 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 Key Establishment Schemes Using Integer Factorization Cryptography
[FIPS 140-2]	FIPS PUB 140-2 Federal Information Processing Standards Publication Security Requirements for Cryptographic Modules May 25, 2001
[FIPS PUB 186-4]	FIPS PUB 186-4 Federal Information Processing Standards Publication Digital Signature Standard (DSS) July 2013
[FIPS PUB 186-5]	FIPS PUB 186-5 Federal Information Processing Standards Publication Digital Signature Standard (DSS) February 3, 2023
[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

9 Annex C: NIAP Technical Decisions

This ST applies the following NIAP Technical Decisions:

Table 20 NIAP Technical Decisions

TD Identifier	TD Name	Protection Profiles	Publication Date	Applicable?
TD0923	NIT Technical Decision: Auditable event for FAU_STG_EXT.1 in FAU_GEN.1.2	CPP_ND_V3.0E	06/25/2025	Yes
TD0921	NIT Technical Decision: Addition of FIPS PUB 186-5 and Correction of Assignment	CPP_ND_V3.0E	06/25/2025	Yes
TD0909	Updates to FCS_SSH_EXT.1.1 App Note in SSH FP 1.0	PKG_SSH_V1.0	04/16/2025	Yes
TD0900	NIT Technical Decision: Clarification to Local Administrator Access in FIA_UIA_EXT.1.3	CPP_ND_V3.0E	02/27/2025	Yes
TD0899	NIT Technical Decision: Correction of Renegotiation Test for TLS 1.2	CPP_ND_V3.0E	03/06/2025	No, SFR not claimed
TD0891	Correlation of Implicitly Satisfied Requirements when CPP_ND_V3.0E is the Base-PP	MOD_MACSEC_V1.0	11/20/2024	Yes
TD0889	Correction For Tests Incorrectly Requiring Group MACsec	MOD_MACSEC_V1.0	10/31/2024	Yes
TD0886	Clarification to FAU_STG_EXT.1 Test 6	CPP_ND_V3.0E	10/16/2024	Yes
TD0884	Expansion of Permitted EtherTypes in FCS_MACSEC_EXT.1.4	MOD_MACSEC_V1.0	10/16/2024	Yes
TD0882	MACsec Data Delay Protection, Key Agreement, and Conditional Support for Group CAK	MOD_MACSEC_V1.0	10/28/2024	Yes
TD0881	Correction to MN Usage for FPT_RPL.1 Test	MOD_MACSEC_V1.0	9/20/2024	Yes
TD0880	Removal of Duplicate Selection in FMT_SMF.1.1	CPP_ND_V3.0E	2024.08.27	Yes
TD0879	Correction of Chapter Headings in CPP_ND_V3.0E	CPP_ND_V3.0E	2024.08.27	Yes
TD0878	Updating FIPS 186-4 to 186-5 in MOD_VPNGW_V1.3	MOD_VPNGW_v1.3	1/30/2025	Yes
TD0870	Security Objectives Rationale for MOD_MACSEC_V1.0	MOD_MACSEC_V1.0	2024.08.08	Yes
TD0868	Clarification of time frames in FCS_IPSEC_EXT.1.7 and FCS_IPSEC_EXT.1.8	CPP_ND_V3.0E	2024.08.27	Yes
TD0840	Alignment of Test 22.1 to FMT_SMF.1/MACSEC	MOD_MACSEC_V1.0	2024.08.14	Yes
TD0838	PPK Configurability in FIA_PSK_EXT.1.1	MOD_VPNGW_v1.3	2024.06.28	Yes

TD0836	NIT Technical Decision: Redundant Requirements in FPT_TST_EXT.1	CPP_ND_V3.0E	2024.04.25	Yes
TD0826	Aligning MOD_MACSEC_V1.0 with CPP_ND_V3.0E	MOD_MACSEC_V1.0	2024.04.25	Yes
TD0824	Aligning MOD_VPNGW 1.3 with NDcPP 3.0E	MOD_VPNGW_v1.3	2024.04.25	Yes
TD0816	Clarity for MACsec Self Test Failure Response	MOD_MACSEC_V1.0	2024.03.22	Yes
TD0811	Correction to Referenced SFR in FIA_PSK_EXT.3 Test	MOD_VPNGW_v1.3	2024.01.02	No, SFR not claimed
TD0803	Clarification for Configurable MACsec CKN Length	MOD_MACSEC_V1.0	10/31/2024	Yes
TD0781	Correction to FIA_PSK_EXT.3 EA for MOD_VPNGW_v1.3	MOD_VPNGW_v1.3	2023.09.11	No, SFR not claimed
TD0777	Clarification to Selections for Auditable Events for FCS_SSH_EXT.1	PKG_SSH_V1.0	2023.08.23	Yes
TD0746	Correction to FPT_RPL.1 Test 25	MOD_MACSEC_V1.0	2023.05.29	Yes
TD0732	FCS_SSHS_EXT.1.3 Test 2 Update	PKG_SSH_V1.0	2023.05.19	Yes
TD0728	Corrections to MACSec PP-Module SD	MOD_MACSEC_V1.0	2023.04.03	Yes
TD0695	Choice of 128 or 256 bit size in AES-CTR in SSH Functional Package	PKG_SSH_V1.0	2022.12.14	Yes
TD0682	Addressing Ambiguity in FCS_SSHS_EXT.1 Tests	PKG_SSH_V1.0	2022.12.13	Yes

10 Annex D: Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly What's New in Cisco Product Documentation, which also lists all new and revised Cisco technical documentation at:

With CCO login: http://www.cisco.com/en/US/partner/docs/general/whatsnew/whatsnew.html

Without CCO login: http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html

Subscribe to the What's New in Cisco Product Documentation as a Really Simple Syndication (RSS) feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service and Cisco currently supports RSS version 2.0.

You can access the most current Cisco documentation on the World Wide Web at the following sites:

http://www.cisco.com

10.1 Document Feedback

If you are reading Cisco product documentation on the World Wide Web, you can submit technical comments electronically. Click Feedback in the toolbar and select Documentation. After you complete the form, click Submit to send it to Cisco.

You can e-mail your comments to bug-doc@cisco.com.

To submit your comments by mail, for your convenience many documents contain a response card behind the front cover. Otherwise, you can mail your comments to the following address:

Cisco Systems, Inc., Document Resource Connection
170 West Tasman Drive
San Jose, CA 95134-9883

We appreciate your comments.

10.2 Obtaining Technical Assistance

Cisco provides Cisco.com as a starting point for all technical assistance. Customers and partners can obtain documentation, troubleshooting tips, and sample configurations from online tools. For Cisco.com registered users, additional troubleshooting tools are available from the TAC website.

Cisco.com is the foundation of a suite of interactive, networked services that provides immediate, open access to Cisco information and resources at any time, from anywhere in the world. This highly integrated Internet application is a powerful, easy-to-use tool for doing business with Cisco.

Cisco.com provides a broad range of features and services to help customers and partners streamline business processes and improve productivity. Through Cisco.com, you can find information about Cisco and our networking solutions, services, and programs. In addition, you can resolve technical issues with online technical support, download and test software packages, and order Cisco learning materials and merchandise. Valuable online skill assessment, training, and certification programs are also available.

Customers and partners can self-register on Cisco.com to obtain additional personalized information and services. Registered users can order products, check on the status of an order, access technical support, and view benefits specific to their relationships with Cisco.

To access Cisco.com, go to the following website:

http://www.cisco.com