Cisco Catalyst 9200/9200L Series Switches running IOS-XE 17.9

Common Criteria Security Target

Version: 0.5
Date: July 24, 2023

Table of Contents

List of Tables

List of Figures

DOCUMENT INTRODUCTION

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

This document provides the basis for an evaluation of a specific Target of Evaluation (TOE), the Cisco Catalyst 9200/9200L Series Switches running IOS-XE 17.9. This Security Target (ST) defines a set of assumptions about the aspects of the environment, a list of threats that the product intends to counter, a set of security objectives, a set of security requirements, and the IT security functions provided by the TOE, which meet the set of requirements. In this document, Administrators of the TOE will be referred to as Administrators, Authorized Administrators, TOE Administrators, semi-privileged, privileged Administrators, and security Administrators.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Date	Version	Update
2022-09-06	0.1	Initial Draft
2022-10-05	0.2	Updates
2023-06-16	0.3	Updates from Testing
2023-07-05	0.4	Updates for check out package
2023-07-24	0.5	Final Updates

1   Security Target Introduction

The Security Target (ST) contains the following sections:

- Security Target Introduction [Section 1]
- Conformance Claims [Section 2]
- Security Problem Definition [Section 3]
- Security Objectives [Section 4]
- Information Technology (IT) Security Requirements [Section 5]
- Target of Evaluation (TOE) Summary Specification [Section 6]
- Annex A: Key Zeroization (Section 7)
- Annex B: NIAP Technical Decisions (Section 8)
- Annex C: Acronyms (Section 9)
- Annex D: Terminology (Section 10)
- Annex E: References (Section 11)

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 the TOE.

Table 1 ST and TOE Identification

Name	Description
ST Title	*Cisco Catalyst 9200/9200L Series Switches running IOS-XE 17.9 Common Criteria Security Target*
ST Version	0.5
Publication Date	July 24, 2023
Vendor and ST Author	Cisco Systems, Inc.
TOE Reference	Cisco Catalyst 9200/9200L Series Switches running IOS-XE 17.9
TOE Hardware Models	Catalyst 9200 and Catalyst 9200L
TOE Software Version	IOS-XE 17.9
Keywords	Audit, Authentication, Encryption, MACsec, Network Device, Secure Administration

1.2   TOE Overview

The TOE is the Cisco Catalyst 9200/9200L Series Switches all running Internetworking Operating System (IOS)-XE 17.9. The TOE is a purpose-built, switching and routing platform with Open System Interconnection (OSI) Layer2 and Layer3 traffic filtering capabilities. The TOE also supports Media Access Control Security (MACsec) encryption for switch-to-switch (inter-network device) security. The TOE includes the hardware models as defined in Table 3 below.

1.2.1   TOE Product Type

The Cisco Catalyst 9200/9200L Series Switches are switching and routing platforms that provide connectivity and security services, including MACsec encryption, on a single, secure device. These switches offer broadband speeds and simplified management to small businesses, enterprise small branch, and teleworkers.

The TOE is a network device that includes MACsec encryption as defined in NDcPP v2.2e[1] and MACsec EP v1.2[2]. The TOE is comprised of both hardware and software. The hardware is the Catalyst 9200 and Catalyst 9200L switches as described in section 1.6 below. The software is the Cisco IOS-XE 17.9.

---

[1] *collaborative Protection Profile for Network Devices Version 2.2e*
[2] *Extended Package for MACsec Ethernet Encryption Version 1.2*

The Cisco Catalyst 9200/9200L Series Switches are single-device security and switching solutions for protecting the network.

1.3 Supported non-TOE Hardware/ Software/ Firmware

The TOE supports the following hardware, software, and firmware components in its operational environment. Each component is identified as being required or not based on the claims made in this ST. All environment components listed in Table 2 below are supported by all TOE evaluated configurations.

Table 2 IT Environment Components

Component	Required	Usage/Purpose Description for TOE performance
Audit (syslog) Server	Yes	This includes any syslog server to which the TOE transmits syslog messages over a secure Internet Protocol security (IPsec) trusted channel.
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
Management Workstation with Secure Shell v2 (SSHv2) client	Yes	This includes any IT Environment Management workstation that is used by the TOE Administrator to support TOE administration using SSHv2 protected channels. Any SSH client that supports SSHv2 may be used
Remote Authentication Dial-In User Service (RADIUS) Authentication, Authorization, and Accounting (AAA) Server	Yes	This includes any IT environment RADIUS AAA server that provides authentication services to TOE Administrators over a secure IPsec trusted channel.
MACsec Peer	Yes	This includes any MACsec peer with which the TOE participates in MACsec communications. MACsec Peer may be any device that supports MACsec communications
Certification Authority (CA)	Yes	This includes any IT Environment CA on the TOE network. The CA can be used to provide the TOE with a valid certificate during certificate enrolment as well as validating a certificate

1.4 TOE Description

This section provides an overview of the TOE, the Catalyst 9200/9200L Series Switches. The TOE is comprised of both software and hardware. The hardware is comprised of the models described in section 1.6 below. The software is comprised of the Universal Cisco IOS-XE 17.9.

Hardware models only vary in component characteristics. These characteristics affect non-security relevant functions, such as throughput and amount of storage. Since there is no security relevant impact due to differing components, equivalence between all switch models is claimed.

Primary features of the Catalyst 9200/9200L Series Switches include the following:

- Central processor that supports all system operations
- Dynamic memory, used by the central processor for all system operations
- Central Processing Unit (CPU) complex with 8-GigaBytes (GB) memory, 16-GB of flash, and an external Universal Serial Bus (USB) 3.0 Solid State Drive (SSD) pluggable storage slot (delivering 120-GB of storage with an optional SSD drive)
- Serial Advanced Technology Attachment (SATA) SSD local storage
- Flash memory Electrically Erasable Programmable Read-Only Memory (EEPROM), used to store the Cisco IOS-XE image (binary program)
- Non-volatile Read Only Memory (ROM) is used to store the bootstrap program and power-on diagnostic programs
- Non-volatile Random Access Memory (NVRAM) is used to store switch configuration parameters that are used to initialize the system at start-up.

- Physical network interfaces (minimally two) (e.g., Registered Jack (RJ-45) serial and standard 10/100/1000 Ethernet ports). The number of network interface ports varies by model
- Dedicated management port on the switch, RJ-45 console port, and a USB mini-Type B console connection
- Resiliency with Field Replaceable Units (FRU) and redundant power supply, fans, and modular uplinks

Cisco IOS-XE is a Cisco-developed highly configurable proprietary operating system that provides for efficient and effective routing and switching. Although IOS-XE performs many networking functions, this evaluation only addresses the functions that provide for the security of the TOE itself as described in section 1.7 below.

Figure 1 below depicts a typical TOE deployment with a single instance of the TOE.
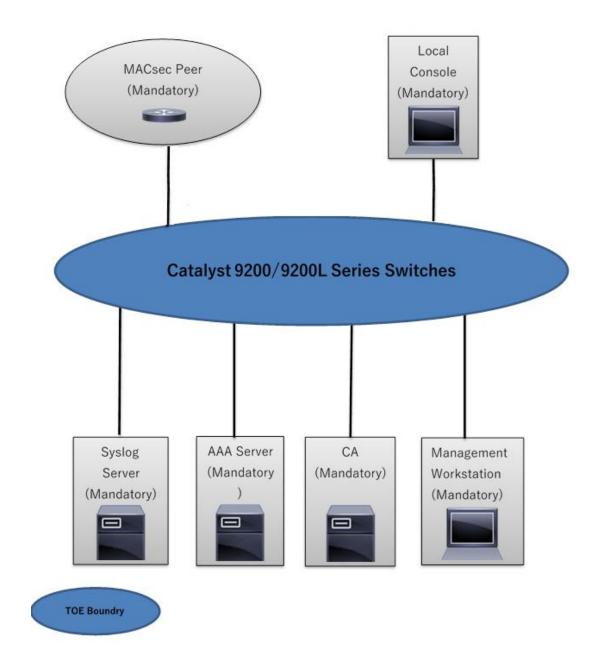
Figure 1 TOE Example Deployment

Figure 1 above includes the following devices, noting the TOE is only the Catalyst 9200/9200L Series Switches and only one TOE device is required for the deployment of the TOE in the evaluated configuration:

- Identifies the TOE Models
  - Catalyst 9200/9200L Series Switches running Cisco IOS-XE 17.9

- Identifies the following IT entities that are in the TOE Operational Environment:
    - o Syslog (audit) Server with a secure connection using IPsec
    - o Local Console to support local Administration (direct connection)
    - o Management Workstation to support remote Administration with a secure connection using SSHv2 Client
    - o RADIUS AAA Server for remote authentication with a secure connection using IPsec
    - o MACsec Peer with a secure connection using MACsec
    - o CA for X509 certificate validation

1.5   TOE Evaluated Configuration

The TOE consists of a physical device, switch, and the Cisco IOS-XE 17.9 software. 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 switch configuration will determine how traffic flows received on an interface will be handled. Typically, packet flows are passed through the internet working device and forwarded to their configured destination.

In addition, if the Catalyst 9200/9200L Series Switches are to be remotely administered, then the management workstation must be connected to an internal network. SSHv2 is used to securely connect to the switch. A syslog server is used to store audit records, where IPsec is used to secure the transmission of the records. If these servers are used, they must be attached to the internal (trusted) network. The internal (trusted) network is meant to be separated effectively from unauthorized individuals and user traffic. The internal (trusted) network is in a controlled environment where implementation of security policies can be enforced.

1.6   Physical Scope of the TOE

The TOE is a hardware and software solution that makes up the switch models as follows: Catalyst 9200/9200L Series Switches running Cisco IOS-XE 17.9. The network, on which they reside, is considered part of the environment. The TOE guidance documentation that is considered to be part of the TOE can be found listed in the Catalyst 9200/9200L Series Switches Common Criteria Operational User Guidance and Preparative Procedures document and are downloadable from the https://software.cisco.com/software/csws/ws/platform/home?locale=en_US# web site. The TOE is comprised of the following physical specifications as described in Table 3 below.

Table 3 Hardware Models and Specifications

Hardware Model	Picture and Dimensions	Specifications
9200 models: C9200-24T C9200-48T C9200-24P C9200-48P C9200-24PB C9200-48PB C9200-48PL C9200-24PXG C9200-48PXG ***With the following network modules:*** C9200-NM-4G C9200-NM-4X C9200-NM-2Y C9200-NM-2Q	**Dimensions (Height (H) x Width (W) x Depth (D)):** 1.73 x 17.5 x 13.8 inches (in) 11.02 - 12.12 pounds (lbs)	**Software:** IOS-XE 17.9 **Processor:** Xilinx ZU3EG (ARM Cortex-A53) **Power Supply:** The internal power supply is an auto-ranging unit. It supports input voltages between 100 and 240 Volts of Alternating Current (VAC). Use the supplied Alternating Current (AC) power cord to connect the AC power connector to an AC power outlet **Interfaces:** 1000BASE-T ports: RJ-45 connectors, 4-pair Category 5E Unshielded Twisted Pair (UTP) cabling 1000BASE-T Small Form Factor Pluggable (SFP)-based ports: RJ-45 connectors, 4-pair Cat 5E UTP cabling 100BASE-FX, 1000BASE-SX, -LX/LH, -ZX, -BX10, dense wavelength-division multiplexing (DWDM) and Coarse Wavelength-Division Multiplexing (CWDM) SFP transceivers: Lucent Connector (LC) fiber connectors (single-mode or multimode fiber) 10GBASE-SR, LR, LRM (only C9200), ER, ZR, DWDM SFP+ transceivers: LC fiber connectors (single-mode or multimode fiber) SFP+ connector Cisco StackWise-160/80 stacking ports: copper-based Cisco StackWise cabling Ethernet management port: RJ-45 connectors, 4-pair Cat 5 UTP cabling Management console port: RJ-45-to-DB9 cable for PC connections, USB-C adaptor, USB adaptor
9200L models: C9200L-24P-4G C9200L-24P-4X C9200L-24T-4G C9200L-24T-4X C9200L-48P-4G C9200L-48P-4X C9200L-48T-4G C9200L-48T-4X C9200L-48PL-4G C9200L-48PL-4X C9200L-24PXG-2Y C9200L-48PXG-2Y C9200L-24PXG-4X C9200L-48PXG-4X	**Dimensions (H x W x D):** C9200-L-24P-4G, C9200L-24P-4X, C9200L-24T-4G, C9200L-24T-4X, C9200L-48P-4G, C9200L-48P-4X, C9200L-48T-4G, C9200L-48T-4X, C9200L-48PL-4G, C9200L-48PL-4X: 1.72 x 17.5 x 13.8 in 9.59 - 10.58 lbs C9200L-24PXG-2Y, C9200L-48PXG-2Y, C9200L-24PXG-4X, C9200L-48PXG-4X: 1.73 x 17.5 12 - 12.6 lbs	**Software:** IOS-XE 17.9 **Processor:** Xilinx ZU3EG (ARM Cortex-A53) **Power Supply:** The internal power supply is an auto-ranging unit. It supports input voltages between 100 and 240 Volts of Alternating Current (VAC). Use the supplied Alternating Current (AC) power cord to connect the AC power connector to an AC power outlet **Interfaces:** 1000BASE-T ports: RJ-45 connectors, 4-pair Cat 5E UTP cabling 1000BASE-T SFP-based ports: RJ-45 connectors, 4-pair Cat 5E UTP cabling 100BASE-FX, 1000BASE-SX, -LX/LH, -ZX, -BX10, dense wavelength-division multiplexing (DWDM) and Coarse Wavelength-Division Multiplexing (CWDM) SFP transceivers: LC fiber connectors (single-mode or multimode fiber) SFP+ connector

Hardware Model	Picture and Dimensions	Specifications
	x 13.8 in	Cisco StackWise-160/80 stacking ports: copper-based Cisco StackWise cabling
		Ethernet management port: RJ-45 connectors, 4-pair Cat 5 UTP cabling
		Management console port: RJ-45-toDB9 cable for PC connections, USB-C adaptor, USB adaptor

1.7   Logical Scope of the TOE

The TOE is comprised of the following security features:

- Security Audit
- Cryptographic Support
- Identification and Authentication
- Security Management
- 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 Request for Comments (RFCs) of the NDcPP v2.2e and MACsec EP v1.2 as necessary to satisfy testing/assurance measures prescribed therein.

1.7.1   Security Audit

The Cisco Catalyst 9200/9200L Series Switches provide extensive auditing capabilities. The TOE generates a comprehensive set of audit logs that identify specific TOE operations. For each event, the TOE records the date and time of each event, the type of event, the subject identity, and the outcome of the event.

Auditable events include:

- failure on invoking cryptographic functionality such as establishment, termination and failure of cryptographic session establishments and connections
- creation and update of Secure Association Key
- modifications to the group of users that are part of the Authorized Administrator roles
- all use of the user identification mechanism
- any use of the authentication mechanism
- Administrator lockout due to excessive authentication failures
- any change in the configuration of the TOE
- changes to time
- initiation of TOE update
- indication of completion of TSF self-test
- maximum sessions being exceeded
- termination of a remote session
- attempts to unlock a termination session
- initiation and termination of a trusted channel

The TOE is configured to transmit its audit messages to an external syslog server. Communication with the syslog server is protected using IPsec and the TOE can determine when communication with the syslog server fails. If that should occur, the TOE will store all audit records locally and when the connection to the remote syslog server is restored, all stored audit records will be transmitted to the remote syslog server.

The audit logs can be viewed on the TOE using the appropriate IOS-XE 17.9 commands. The records include the date/time the event occurred, the event/type of event, the user associated with the event, and additional information of the event and its success and/or failure. The TOE does not have an interface to modify audit records, though there is an interface available for the Authorized Administrator to clear audit data stored locally on the TOE.

1.7.2 Cryptographic Support

The TOE provides cryptographic functions to implement IPsec, SSH, and MACsec protocols. The cryptographic algorithm implementation has been validated for CAVP conformance. This includes key generation and random bit generation, key establishment methods, key destruction, and the various types of cryptographic operations to provide AES encryption/decryption, signature verification, hash generation, and keyed hash generation.

The TOE leverages the IOS Common Cryptographic Module (IC2M), firmware version Rel5a (CAVP cert. #A1462). The IOS software calls the IC2M Rel5a cryptographic module that is validated for conformance to the requirements of Federal Information Processing Standards (FIPS) 140-2 Level 1.

The TOE supports MACsec using the proprietary Unified Access Data Plane (UADP) 2.0 Application-Specific Integrated Circuit (ASIC) (CAVP Cert. #4769). The MACsec Controller (MSC) is embedded within the ASICs that are utilized within Cisco hardware platforms.

Refer to Table 4 below for algorithm certificate references.

Table 4 FIPS Algorithm References

SFR	Selection	Algorithm	Implementation	Certificate Number
FCS_CKM.1 – Cryptographic Key Generation	2048 3072	RSA	IC2M	A1462
FCS_COP.1/DataEncryption – AES Data Encryption/Decryption	AES-CBC-128 AES-CBC-256	AES	IC2M	A1462
FCS_COP.1.1(5) Cryptographic Operation (MACsec AES Data Encryption/Decryption)	AES-GCM-128	AES	UADP MSC	AES 4769 [3]
FCS_COP.1.1(5) Cryptographic Operation (MACsec AES Data Encryption/Decryption)	AES-KW 128 bits	AES	IC2M	A1462
FCS_COP.1/SigGen – Cryptographic Operation (Signature Generation and Verification)	2048 3072	RSA	IC2M	A1462
FCS_COP.1/Hash – Cryptographic Operation (Hash Algorithm)	SHA-1 SHA-256 SHA-512	SHS	IC2M	A1462

---

[3] The Tested Environment is Synopsys VCS v2011.12mx-SP1-3

SFR	Selection	Algorithm	Implementation	Certificate Number
FCS_COP.1/KeyedHash – Cryptographic Operation (Keyed Hash Algorithm)	HMAC-SHA-1 HMAC-SHA-256 HMAC-SHA-512	HMAC	IC2M	A1462
FCS_COP.1(1)/KeyedHashCMAC Cryptographic Operation (AES-CMAC Keyed Hash Algorithm)	AES-CMAC 128 bits	AES-CMAC	IC2M	A1462
FCS_RBG_EXT.1– Random Bit Generation	CTR_DRBG (AES) 256 bits	DRBG	IC2M	A1462

The TOE provides cryptographic support for IPsec, which is used to secure sessions between the TOE and remote syslog servers and also between the TOE and remote authentication servers (RADIUS).

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.

The cryptographic services provided by the TOE are described in Table 5 below.

Table 5 TOE Provided Cryptography

Cryptographic Method	Use within the TOE
AES	Used to encrypt IPsec session traffic Used to encrypt SSH session traffic Used to encrypt MACsec traffic
HMAC	Used for keyed hash, integrity services in IPsec and SSH session establishment
DH	Used as the Key exchange method for IPsec and SSH
Internet Key Exchange	Used to establish initial IPsec session
RSA Signature Services	Used in IPsec session establishment Used in SSH session establishment X.509 certificate signing
RSA	Used in IKE protocols peer authentication Used to provide cryptographic signature services Used in Cryptographic Key Generation and Key Establishment
Secure Shell Establishment	Used to establish initial SSH session
SHS	Used to provide IPsec traffic integrity verification Used to provide SSH traffic integrity verification Used for keyed-hash message authentication
ISO/IEC 18031:2011 CTR_CRBG (AES)	Used for random number generation, key generation and seeds to asymmetric key generation Used in IPsec session establishment Used in SSH session establishment Used in MACsec session establishment

The Catalyst 9200/9200L Series Switches contain the processors listed in Table 3 above.

1.7.3   Identification and Authentication

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

The TOE provides authentication services for administrative users to connect to the TOE's secure Command Line Interface (CLI) Administrator interface. The TOE requires Authorized Administrators to authenticate prior to being granted access to any of the management functionality. The TOE can be configured to require a minimum password length of 15 characters as well as mandatory password complexity rules. The TOE provides Administrator authentication against a local user database. Password-based authentication can be performed on the local serial console or SSHv2 interfaces. The SSHv2 interface also supports authentication using SSH keys. The TOE supports 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 connection to the remote authentication server is secured using IPsec.

The TOE also provides authentication failure management when a user attempts to authenticate and enters invalid information. When the threshold for a defined number of failed authentication attempts has exceeded the configured allowable attempts, the account will not be granted access until the time period has elapsed.

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

1.7.4   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 serial console connection. The TOE provides the ability to securely manage:

- Administration of the TOE locally and remotely
- Configuration of warning and consent access banners
- Configuration of session inactivity thresholds
- Updates of the TOE software
- Configuration of authentication failures
- Configuration of the audit functions of the TOE
- Configuration of the cryptographic functionality of the TOE
- Install and manage Pre-Shared Key (PSK)
- Manage the Key Server, Connectivity Association Key (CAK) and MKA participants
- Configure lockout time interval for excessive authentication failures

The TOE supports two separate Administrator roles: non-privileged Administrator and privileged Administrator. Only the privileged Administrator can perform the above security relevant management functions. The privileged Administrator is the Authorized Administrator of the TOE who can enable, disable, determine, and modify the behaviour of the security functions of the TOE as described in this document.

1.7.5   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 can verify any software updates prior to the software updates being installed on the TOE to avoid the installation of unauthorized software.

The TOE detects replay of information received via secure channels (MACsec). The detection is 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 internally maintains the date and time. This date and time information is used as the timestamp that is applied to audit records generated by the TOE. The TOE provides the Authorized Administrators the capability to update the TOE's clock manually to maintain a reliable timestamp.

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

1.7.6   TOE Access

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

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

1.7.7   Trusted path/Channels

The TOE allows a trusted path to be established to itself from remote Administrators over SSHv2 and initiates outbound IPsec trusted channels to transmit audit messages to remote syslog servers. In addition, IPsec is used as a trusted channel between the TOE and the remote authentication servers.

The TOE supports MACsec secured trusted channels between itself and MACsec peers.

1.8 Excluded Functionality

Functionality in Table 6 below is excluded from the evaluation.

Table 6 Excluded Functionality

Excluded Functionality	Exclusion Rationale
Non-FIPS 140-2 mode of operation	This mode of operation includes non-FIPS allowed operations
Telnet	Telnet sends authentication data in plain text. This feature must remain disabled in the evaluated configuration. SSHv2 must be used to secure the trusted path for remote administration for all SSHv2 sessions.
Transport Layer Security (TLS)	TLS is not associated with Security Functional Requirements claimed in [NDcPP] IPsec is used instead.
Hypertext Transfer Protocol (HTTP)	HTTP Is not associated with Security Functional Requirements claimed in [NDcPP] Use tunnelling through IPSEC.
Hypertext Transfer Protocol Secure (HTTPS)	HTTPS is not associated with Security Functional Requirements claimed in [NDcPP] Use tunnelling through IPSEC.

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

2 Conformance Claims

2.1 Common Criteria Conformance Claim

The TOE and ST are compliant with the Common Criteria (CC) Version 3.1, Revision 5, dated: April 2017. The TOE and ST are CC Part 2 extended and CC Part 3 conformant.

2.2 Protection Profile Conformance

The TOE and ST are conformant with the Protection Profiles as listed in Table 7 below. This ST applies the NIAP Technical Decisions described in Table 20 in section 8 below.

Table 7 Protection Profiles

Protection Profile	Version	Date
collaborative Protection Profile for Network Devices (NDcPP)	2.2e	March 23, 2020
Network Device Protection Profile Extended Package MACsec Ethernet Encryption (MACSec EP)	1.2	May 10, 2016

2.2.1 TOE Appropriateness

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

- collaborative Protection Profile for Network Devices Version 2.2e (NDcPP v2.2e)
- Network Device collaborative Protection Profile (NDcPP) Extended Package MACsec Ethernet Encryption, Version 1.2 (MACsec EP v1.2)

2.2.2 TOE Security Problem Definition Consistency

The Assumptions, Threats, and Organization Security Policies included in the Security Target represent the Assumptions, Threats, and Organization Security Policies specified in the NDcPP v2.2e and the MACsec EP v1.2 for which conformance is claimed verbatim. All concepts covered in the Protection Profile and Extended Package Security Problem Definition is included in the Security Target Statement of Security Objectives Consistency.

The Security Objectives included in the Security Target represent the Security Objectives specified in the NDcPP v2.2e and the MACsec EP v1.2, for which conformance is claimed verbatim. All concepts covered in the Protection Profile and Extended Package Statement of Security Objectives is included in the Security Target.

2.2.3 Statement of Security Requirements Consistency

The Security Functional Requirements included in the Security Target represent the Security Functional Requirements specified in the NDcPP v2.2e and the MACsec EP v1.2, for which conformance is claimed verbatim. All concepts covered in the Protection Profile and Extended Package Statement of Security Requirements is included in this Security Target. Additionally, the Security Assurance Requirements included in this Security Target are identical to the Security Assurance Requirements included in the NDcPP v2.2e and the MACsec EP v1.2.

3   Security Problem Definition

This section identifies the following:

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

This document identifies assumptions as A.assumption with "assumption" specifying a unique name. Threats are identified as T.threat with "threat" specifying a unique name. Organizational Security Policies (OSPs) are identified as P.osp with "osp" specifying a unique name.

3.1   Assumptions

The specific conditions listed in the following subsections are assumed to exist in the TOE's environment. These assumptions include both practical realities in the development of the TOE security requirements and the essential environmental conditions on the use of the TOE. Note, the assumption, A.NO_THRU_TRAFFIC_PROTECTION is strike-through since the TOE does provide protection against the traffic that does traverse the TOE, which is countered by the TOE objectives defined in 4.1 Security Objectives for the TOE.

Table 8 TOE Assumptions

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

Assumption	Assumption Definition
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.

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.
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., a 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. Non-validated updates or updates validated using non-secure or weak cryptography leave the update firmware vulnerable to surreptitious alteration.
T.UNDETECTED_ACTIVITY	Threat agents may attempt to access, change, and/or modify the security functionality of the network device without Administrator awareness. This could result in the attacker finding an avenue (e.g., misconfiguration, flaw in the product) to compromise the device and the Administrator would have no knowledge that the device has been compromised.
T.SECURITY_FUNCTIONALITY_COMPROMISE	Threat agents may compromise credentials and device data enabling continued access to the network device and its critical data. The compromise of credentials includes replacing existing credentials with an attacker's credentials, modifying existing credentials, or obtaining the Administrator or device credentials for use by the attacker.
T.PASSWORD_CRACKING	Threat agents may be able to take advantage of weak administrative passwords to gain privileged access to the device. Having privileged access to the device provides the attacker unfettered access to the network traffic and may allow them to take advantage of any trust relationships with other network devices.

Threat	Threat Definition
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	An attacker may modify data transmitted over the MACsec channel in a way that is not detected by the recipient.
T.NETWORK_ACCESS	An attacker may send traffic through the TOE that enables them to access devices in the TOE's Operational Environment without authorization.
T.UNTRUSTED_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.

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 users 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.

4.1 Security Objectives for the TOE

The NDcPP v2.2e does not define any security objectives for the TOE, however the MACsec EP v1.2 includes security objectives listed in Table 11 below specific to MACsec devices.

Table 11 Security Objectives for the TOE

Security Objective and SFR mapping	Security Objective Definition
O.CRYPTOGRAPHIC_FUNCTIONS (FCS_COP.1/DataEncryption, FCS_MACSEC_EXT.2, FCS_MACSEC_EXT.3, FTP_ITC.1, FTP_TRP.1)	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.AUTHENTICATION (FCS_MACSEC_EXT.4, FCS_MKA_EXT.1, FIA_PSK_EXT.1)	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 (CA) with another MACsec peer. MACsec endpoints authenticate each other to ensure they are communicating with an authorized SecY entity (SeY).
O.PORT_FILTERING (FCS_MACSEC_EXT.1, FIA_PSK_EXT.1)	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 source address/port and whether or not the traffic represents valid MACsec frames and MACsec Key Agreement Protocol Data Unit(MKPDU)s.
O.SYSTEM_MONITORING (FAU_GEN.1)	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 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.AUTHORIZED_ADMINISTRATION (FIA_AFL.1, FMT_SMF.1, FPT_CAK_EXT.1, FTP_TRP.1)	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 behaviour. 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.TSF_INTEGRITY (FPT_FLS.1(2)/SelfTest)	To mitigate the security risk that the MACsec device may fail during startup, it is required to shut down in the event that any self-test failures occur during startup. This ensures that the device will only operate when it is in a known state.
O.REPLAY_DETECTION (FPT_RPL.1,)	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 MACsec protocol data units (MPDUs).
O.VERIFIABLE_UPDATES (FPT_TUD_EXT.1)	To ensure the authenticity and integrity of software/firmware updates that are loaded onto the MACsec device, it is necessary to provide a mechanism for validating these updates prior to application. The NDcPP provides methods of update verification; this EP specifically requires that a signature-based mechanism be used at minimum.

4.2 Security Objectives for the Environment

All 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. Note, the environment security objective, OE.NO_THRU_TRAFFIC_PROTECTION is strike-through since the TOE does provide protection against the traffic that does traverse the TOE, which is countered by the TOE objectives defined in 4.1 Security Objectives for the TOE.

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.
~~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 TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are assumed to monitor the revocation status of all certificates in the TOE's trust store and to remove any certificate from the TOE's trust store in case such certificate can no longer be trusted.
OE.UPDATES	The TOE firmware and software is updated by an Administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
OE.ADMIN_CREDENTIALS_SECURE	The Administrator's credentials (private key) used to access the TOE must be protected on any other platform on which they reside.
OE.RESIDUAL_INFORMATION	The Security Administrator ensures that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment.

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 and claimed PP/EP:

- Unaltered SFRs are stated in the form used in [CC2] or their extended component definition (ECD)
- Refinement made by PP author: Indicated with **bold text** and ~~strikethroughs~~
- Selection wholly or partially completed in the PP: the selection values (i.e., the selection values adopted in the PP or the remaining selection values available for the ST) are indicated with underlined text
    - e.g., "[selection: *disclosure, modification, loss of use*]" in [CC2] or an ECD might become "disclosure" (completion) or "[selection: disclosure, modification]" (partial completion) in the PP
- Assignment wholly or partially completed in the PP: indicated with *italicized text*
- Assignment completed within a selection in the PP: the completed assignment text is indicated with italicized and underlined text
    - e.g., "[selection: *change_default*, *query*, *modify*, *delete*, [*assignment: other operations*]]" in [CC2] or an ECD might become "change default, select tag" (completion of both selection and assignment) or "[selection: change_default, *select_tag*, *select_value*]" (partial completion of selection, and completion of assignment) in the PP;
- Iteration: indicated by adding a string starting with "/" (e.g., "FCS_COP.1/Hash")

Extended SFRs are identified by having a label "EXT" at the end of the SFR name.

Formatting conventions outside of operations and iterations matches the formatting specified within the NDcPP v2.2e and MACsec EP v1.2.

The following conventions were used to resolve conflicting SFRs between NDcPP v2.2e and MACsec EP v1.2:

- All SFRs from MACsec EP reproduced as-is
- SFRs that appear in both NDcPP and MACsec EP are modified based on instructions specified in the MACsec EP

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.2	User Identity Association
	FAU_STG_EXT.1	Protected Audit Event Storage
FCS: Cryptographic support	FCS_CKM.1	Cryptographic Key Generation (for asymmetric keys)
	FCS_CKM.2	Cryptographic Key Establishment
	FCS_CKM.4	Cryptographic Key Destruction
	FCS_COP.1/DataEncryption	Cryptographic Operation (AES Data Encryption/ Decryption)
	FCS_COP.1/SigGen	Cryptographic Operation (Signature Generation and Verification)

Class Name	Component Identification	Component Name
	FCS_COP.1/Hash	Cryptographic Operation (Hash Algorithm)
	FCS_COP.1/KeyedHash	Cryptographic Operation (Keyed Hash Algorithm)
	FCS_COP.1(1)/KeyedHashCMAC	KeyedHashCMAC Cryptographic Operation (AES-CMAC Keyed Hash Algorithm)
	FCS_COP.1(5) Cryptographic Operation (MACsec Data Encryption/Decryption)	Cryptographic Operation (MACsec Data Encryption/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_SSHS_EXT.1	SSH Server Protocol
	FCS_RBG_EXT.1	Random Bit Generation
FIA: Identification and authentication	FIA_AFL.1	Authentication Failure Handling
	FIA_PMG_EXT.1	Password Management
	FIA_PSK_EXT.1 Extended	Pre-Shared Key Composition
	FIA_UIA_EXT.1	User Identification and Authentication
	FIA_UAU_EXT.2	Password-based Authentication Mechanism
	FIA_UAU.7	Protected Authentication Feedback
	FIA_X509_EXT.1/Rev	X.509 Certificate Validation
	FIA_X509_EXT.2	X.509 Certificate Authentication
	FIA_X509_EXT.3	X.509 Certificate Requests
FMT: Security management	FMT_MOF.1/ManualUpdate	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_SMR.2	Restrictions on Security Roles
FPT: Protection of the TSF	FPT_APW_EXT.1	Protection of Administrator Passwords
	FPT_CAK_EXT.1	Protection of CAK Data
	FPT_FLS.1	SelfTest Failure with Preservation of Secure State
	FPT_RPL.1	Replay Detection
	FPT_SKP_EXT.1	Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)
	FPT_STM_EXT.1	Reliable Time Stamps
	FPT_TST_EXT.1	TSF Testing
	FPT_TUD_EXT.1	Trusted Update
FTA: TOE Access	FTA_SSL_EXT.1	TSF-initiated Session Locking
	FTA_SSL.3	TSF-initiated Termination
	FTA_SSL.4	User-initiated Termination
	FTA_TAB.1	Default TOE Access Banners
FTP: Trusted path/channels	FTP_ITC.1	Inter-TSF trusted channel
	FTP_TRP.1/Admin	Trusted Path

5.2.1   Security audit (FAU)

5.2.1.1   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 <u>not specified</u> level of audit; and
  c) *All administrative actions comprising:*
     • *Administrative login and logout (name of user account shall be logged if individual user accounts are required for Administrators).*

- *Changes to TSF data related to configuration changes (in addition to the information that a change occurred it shall be logged what has been changed).*
- *Generating/import of, changing, or deleting of cryptographic keys (in addition to the action itself a unique key name or key reference shall be logged).*
- *Resetting passwords (name of related user account shall be logged).*
- [*no other actions*];

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, 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*.

Table 14 Auditable Events

SFR	Auditable Event	Additional Audit Record Contents
FAU_GEN.1	None.	None.
FAU_GEN.2	None.	None.
FAU_STG_EXT.1	None.	None.
FCS_CKM.1	None.	None.
FCS_CKM.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(1)/KeyedHashCMAC	None	None
FCS_COP.1(5) Cryptographic Operation (MACsec Data Encryption/Decryption)	None.	None.
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.4.4	Creation of Connectivity Association	Connectivity Association Key Names
FCS_MACSEC_EXT.3.1	Creation and update of Secure Association Key	Creation and update times
FCS_SSHS_EXT.1	Failure to establish an SSH session	Reason for failure.
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).
	Administrator lockout due to excessive authentication failures	None.
FIA_PMG_EXT.1	None.	None.
FIA_PSK_EXT.1	None.	None.
FIA_UIA_EXT.1	All use of the identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UAU_EXT.2	All use of the identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UAU.7	None.	None.
FIA_X509_EXT.1/Rev	Unsuccessful attempt to validate a certificate 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.
FIA_X509_EXT.3	None.	None.
FMT_MOF.1/ ManualUpdate	Any attempt to initiate a manual update	None.
FMT_MTD.1/CoreData	None	None
FMT_MTD.1/CryptoKeys	None	None
FMT_SMF.1	All management activities of TSF data.	None.
FMT_SMR.2	None.	None.
FPT_FLS.1	None.	None.

SFR	Auditable Event	Additional Audit Record Contents
FPT_SKP_EXT.1	None.	None.
FPT_APW_EXT.1	None.	None.
FPT_RPL.1	Detected replay attempt	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_TUD_EXT.1	Initiation of update; result of the update attempt (success and failure)	None.
FTA_SSL_EXT.1	The termination of a local session by the session locking mechanism.	None.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	None.
FTA_SSL.4	The termination of an interactive session.	None.
FTA_TAB.1	None.	None.
FTP_ITC.1	Initiation of the trusted channel. Termination of the trusted channel. Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempt
FTP_TRP.1/Admin	Initiation of the trusted path. Termination of the trusted path. Failures of the trusted path functions.	None.

5.2.1.2   FAU_GEN.2 User Identity Association

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

5.2.1.3   FAU_STG_EXT.1 Protected Audit Event Storage

FAU_STG_EXT.1.1 The TSF shall be able to transmit the generated audit data to an external IT entity using a trusted channel according to FTP_ITC.1.

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

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

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

5.2.2   Cryptographic Support (FCS)

5.2.2.1   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 or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3;

- 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].

5.2.2.2   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: [

- RSA-based key establishment schemes that meet the following: RSAES-PKCS1-v1_5 as specified in Section 7.2 of RFC 3447, "Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1";
- 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].

5.2.2.3   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-pass*] overwrite consisting of *[zeroes]*
    ]

that meets the following: *No Standard*.

5.2.2.4   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] *mode* and cryptographic key sizes [128 bits, 256 bits] that meet the following: *AES as specified in ISO 18033-3,* **[**CBC as specified in ISO 10116].

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

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

- RSA Digital Signature Algorithm and cryptographic key sizes (*modulus*) [*2048 bits, 3072 bits*],

]
that meet the following: [

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

].

5.2.2.6 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-512] and cryptographic key sizes [assignment: cryptographic key sizes] and **message digest sizes** [160, 256, 512] **bits** that meet the following: *ISO/IEC 10118-3:2004*.

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

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

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

FCS_COP.1.1(1)/KeyedHash:CMAC Refinement: The TSF shall perform **keyed-hash message authentication** in accordance with a specified cryptographic algorithm **[AES-CMAC]** and cryptographic key sizes [128 bits**]** and message digest size **of 128** bits that meets **NIST SP800-38B.**

5.2.2.9 FCS_COP.1(5) Cryptographic Operation (MACsec Data Encryption/Decryption)

FCS_COP.1.1(5) Refinement: The TSF shall perform **encryption/decryption** in accordance with a specified cryptographic algorithm AES used in **AES Key Wrap, GCM** and cryptographic key sizes [128 bits] that meet the following: **AES as specified in ISO 18033-3, AES Key Wrap as specified in NIST SP800-38F, GCM as specified in ISO 19772**.

5.2.2.10 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[4] as defined by RFC 4303 using the cryptographic algorithms [AES-CBC-128 (RFC3602), AES-CBC-256 (RFC3602)] together with a Secure Hash Algorithm (SHA)-based HMAC [HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-512].

FCS_IPSEC_EXT.1.5 The TSF shall implement the protocol: [
- IKEv2 as defined in RFC 5996 and [with no support for NAT traversal], 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-256 (specified in RFC 3602)].

FCS_IPSEC_EXT.1.7 The TSF shall ensure that [

---

[4] ESP – Encapsulating Security Protocol

- IKEv2 SA lifetimes can be configured by a Security Administrator based on
[
  - o length of time, where the time values can be configured within [*2 minutes to 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
[
  - o number of bytes
  - o length of time, where the time values can be configured within [*2 minutes to 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)*] bits.

FCS_IPSEC_EXT.1.10 The TSF shall generate nonces used in [IKEv2] exchanges of length [
- at least 128 bits in size and at least half the output size of the negotiated pseudorandom function (PRF) hash
].

FCS_IPSEC_EXT.1.11 The TSF shall ensure that IKE protocols implement DH Group(s) [[14 (2048-bit MODP)] according to RFC 3526.
].

FCS_IPSEC_EXT.1.12 The TSF shall be able to ensure by default that the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the [IKEv2 IKE_SA] connection is greater than or equal to the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the [IKEv2 CHILD_SA] connection.

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

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

5.2.2.11 FCS_MACSEC_EXT.1 MACsec

FCS_MACSEC_EXT.1.1 The TSF shall implement MACsec in accordance with IEEE Standard 802.1AE-2006.

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 a MACsec Protocol Data Unit (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 (PAE EtherType 88-8E), MACsec frames (EtherType 88-E5), and MAC control frames (EtherType is 88-08) and shall discard others.

5.2.2.12 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 Secure Association Key (SAK).

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

5.2.2.13 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 nonce for the derivation of SAKs using the TOE's random bit generator as specified by FCS_RBG_EXT.1.

5.2.2.14 FCS_MACSEC_EXT.4 MACsec Key Usage

FCS_MACSEC_EXT.4.1 The TSF shall support peer authentication using pre-shared keys, [no other methods].

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

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 Security Association Key (SAK)s that are defined by the key derivation function using the CAK as input data (per 802.1X, section 9.8.1).

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

5.2.2.15 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 enable data delay protection for MKA that ensures data frames protected by MACsec are not delayed by more than 2 seconds.

FCS_MKA_ EXT.1.3 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.1.4 The TSF shall provide the ability to derive an Integrity Check Value Key (ICK) from a CAK using a KDF.

FCS_ MKA_EXT.1.5 The TSF shall enforce an MKA Lifetime Timeout limit of 6.0 seconds and MKA Bounded Hello Time limit of 0.5 seconds.

FCS_MKA_EXT.1.6 The Key Server shall refresh a SAK when it expires. The Key Server shall distribute a SAK by [pairwise CAKs]. ~~If group CAK is selected, then the Key Server shall distribute a group CAK by [selection: a group CAK, pairwise CAKs, pre-shared key].~~ If pairwise CAK is selected, then the pairwise CAK shall be [pre-shared key]. The Key Server shall refresh a CAK when it expires.

FCS_MKA_EXT.1.7 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.1.8 The TSF shall validate MKPDUs according to 802.1X, 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 is not a multiple of 4 octets long.
   d) 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.
   e) 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 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 802.1X, section 9.4.1 shall be decoded as specified in 802.1X, section 11.11.4.

5.2.2.16 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*] platform 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.

5.2.2.17 FCS_SSHS_EXT.1 SSH Server Protocol

FCS_SSHS_EXT.1.1 The TSF shall implement the SSH protocol that complies with: RFCs 4251, 4252, 4253, 4254, [6668, 8308 section 3.1, 8332].

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

FCS_SSHS_EXT.1.3 The TSF shall ensure that, as described in RFC 4253, packets greater than [*33,038*] bytes in an SSH transport connection are dropped.

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

FCS_SSHS_EXT.1.5 The TSF shall ensure that the SSH public-key based authentication implementation uses [rsa-sha2-256, rsa-sha2-512] as its public key algorithm(s) and rejects all other public key algorithms.

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

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

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

5.2.3 Identification and authentication (FIA)

5.2.3.1 FIA_AFL.1 Authentication Failure Handling

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

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been [met], the TSF shall [prevent the offending remote administrator from successfully authenticating until an Administrator defined time period has elapsed].

5.2.3.2 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: ["!", "@", "#", "$", "%", "^", "&", "*", "(", ")"[*Additional Special Characters listed in Table 15*]]*;*

Table 15. Additional Password Special Characters

Special Character	Name
	Space
;	Semicolon
:	Colon
"	Double Quote
'	Single Quote
|	Vertical Bar
+	Plus
-	Minus
=	Equal Sign

.	Period
,	Comma
/	Slash
\	Backslash
<	Less Than
>	Greater Than
_	Underscore
`	Grave accent (backtick)
~	Tilde
{	Left Brace
}	Right Brace

b) Minimum password length shall be configurable to between [*1*] and [*127*] characters.

5.2.3.3   FIA_PSK_EXT.1 Extended: Pre-Shared Key Composition

FIA_PSK_EXT.1.1 The TSF shall use pre-shared keys for MKA as defined by IEEE 802.1X, [*IPsec protocols*].

FIA_PSK_EXT.1.2 The TSF shall be able to [accept] bit-based pre-shared keys.

5.2.3.4   FIA_UIA_EXT.1 User Identification and Authentication

FIA_UIA_EXT.1.1 The TSF shall allow the following actions prior to requiring the non-TOE entity to initiate the identification and authentication process:
- Display the warning banner in accordance with FTA_TAB.1;
- [no other actions].

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

5.2.3.5   FIA_UAU_EXT.2 Password-based Authentication Mechanism

FIA_UAU_EXT.2.1 The TSF shall provide a local [password-based, SSH public key-based [*remote password-based authentication via RADIUS*]] authentication mechanism to perform local administrative user authentication.

5.2.3.6   FIA_UAU.7 Protected Authentication Feedback

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

5.2.3.7 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 certification 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 [a 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 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.

5.2.3.8 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 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].

5.2.3.9 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.

5.2.4 Security Management (FMT)

5.2.4.1 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*.

5.2.4.2 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*.

5.2.4.3 FMT_MTD.1/CryptoKeys Management of TSF Data

FMT_MTD.1.1/CryptoKeys The TSF shall restrict the ability to manage the *cryptographic keys to Security Administrators*.

5.2.4.4 FMT_SMF.1 Specification of Management Functions

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

- *Ability to administer the TOE locally and remotely;*
- *Ability to configure the access banner;*
- *Ability to configure the session inactivity time before session termination or locking;*
- *Ability to update the TOE, and to verify the updates using [digital signature] capability prior to installing those updates;*
- *Ability to configure the authentication failure parameters for FIA_AFL.1;*
- *Generate a PSK-based CAK and install it in the device;*
- *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())];*
- *Specify a lifetime of a CAK;*
- *Enable, disable, or delete a PSK-based CAK using [CLI management commands];*
- *Configure the number of failed Administrator authentication attempts that will cause an account to be locked out [Configure the time interval for administrator lockout due to excessive authentication failures];*
- [
  - o Ability to modify the behaviour of the transmission of audit data to an external IT entity;
  - o Ability to manage the cryptographic keys;
  - o Ability to configure the cryptographic functionality;
  - o Ability to configure thresholds for SSH rekeying;
  - o Ability to configure the lifetime for IPsec SAs;
  - o Ability to set the time which is used for time-stamps;
  - o Ability to configure the reference identifier for the peer;
  - o Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors;
  - o Ability to import X.509v3 certificates to the TOE's trust store;
  - o Ability to manage the trusted public keys database
  ]
].

5.2.4.5 FMT_SMR.2 Restrictions on Security Roles

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

FMT_SMR.2.2 The TSF shall be able to associate users with roles.

FMT_SMR.2.3 The TSF shall ensure that the conditions
- *The Security Administrator role shall be able to administer the TOE locally;*
- *The Security Administrator role shall be able to administer the TOE remotely*
are satisfied.

5.2.5   Protection of the TSF (FPT)

5.2.5.1   FPT_APW_EXT.1: 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.

5.2.5.2   FPT_CAK_EXT.1 Protection of CAK Data

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

5.2.5.3   FPT_FLS.1 (2)/SelfTest Failure with Preservation of Secure State

FPT_FLS.1.1(2)/SelfTest Refinement: The TSF shall **shut down** 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**.

5.2.5.4   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.

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

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

5.2.5.6   FPT_STM.1 Reliable time stamps

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

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

5.2.5.7   FPT_TST_EXT.1: TSF Testing

FPT_TST_EXT.1.1 The TSF shall run a suite of the following self-tests [during initial start-up (on power on), periodically during normal operation] to demonstrate the correct operation of the TSF: [
- *AES Known Answer Test*
- *HMAC Known Answer Test*
- *RNG/DRBG Known Answer Test*
- *SHA-1/256/512 Known Answer Test*
- *RSA Signature Known Answer Test (both signature/verification)*
- *Software Integrity Test*
].

5.2.5.8   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] prior to installing those updates.

5.2.6   TOE Access (FTA)

5.2.6.1   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.

5.2.6.2   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*.

5.2.6.3   FTA_SSL.4   User-initiated Termination

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

5.2.6.4   FTA_TAB.1 Default TOE Access Banners

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

5.2.7   Trusted Path/Channels (FTP)

5.2.7.1   FTP_ITC.1   Inter-TSF trusted channel

FTP_ITC.1.1 Refinement: The TSF shall **be capable of using [**IPsec, MACsec**] to** provide a trusted communication channel between itself and **authorized IT entities supporting the following capabilities: audit server,** [authentication server*, [MACsec peers]*] that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from disclosure and detection of modification of the channel data.

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

FTP_ ITC.1.3 The TSF shall initiate communication via the trusted channel for [
- *external audit server using IPsec*
- *remote AAA servers using IPsec*

- *MACsec peers using MACsec*

].

5.2.7.2   FTP_TRP.1 Trusted Path

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

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

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

5.3   TOE SFR Dependencies Rationale for SFRs Found in NDcPP v2.2e

The Security Functional Requirements (SFRs) in this Security Target represent the SFRs identified in the NDcPP v2.2e and MACsec EP v1.2. As such, the NDcPP v2.2e and MACsec EP v1.2 SFR dependency rationale is deemed acceptable since the PP itself has been validated.

5.4   Security Assurance Requirements

5.4.1   SAR[5] Requirements

The TOE assurance requirements for this ST are taken directly from the NDcPP v2.2e and MACsec EP v1.2, which are derived from Common Criteria Version 3.1, Revision 5, dated April 2017. The assurance requirements are summarized in Table 16 below.

Table 16 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
Tests (ATE)	ATE_IND.1	Independent testing - conformance
Vulnerability Assessment (AVA)	AVA_VAN.1	Vulnerability survey

5.4.2   Security Assurance Requirements Rationale

The Security Assurance Requirements (SARs) in this Security Target represent the SARs identified in the NDcPP v2.2e and MACsec EP v1.2. As such, the NDcPP v2.2e and MACsec EP v1.2 SAR rationale is deemed acceptable since the PP itself has been validated.

---

[5] SAR – Security Assurance Requirements

5.5 Assurance Measures

The TOE satisfies the identified assurance requirements. This section identifies the Assurance Measures applied by Cisco to satisfy the assurance requirements. Assurance measures are provided in Table 17 below.

Table 17 Assurance Measures

Component	How requirement will be met
Security Target (ASE) ASE_CCL.1 ASE_ECD.1 ASE_INT.1 ASE_OBJ.1 ASE_REQ.1 ASE_SPD.1 ASE_TSS.1	Section 2 of this ST includes the TOE and ST conformance claim to CC Version 3.1, Revision 5, dated: April 2017, CC Part 2 extended and CC Part 3 conformant, NDcPP v2.2e and MACsec EP v1.2 and the rationale of how TOE provides all of the functionality at a level of security commensurate with that identified in NDcPP v2.2e and MACsec EP v1.2. Section 2 also includes the consistency rationale for the TOE Security Problem Definition and the Security Requirements to include the extended components definition.
ADV_FSP.1	The functional specification describes the external interfaces of the TOE, such as the means for a user to invoke a service and the corresponding response of those services. The description includes the interface(s) that enforces a security functional requirement, the interface(s) that supports the enforcement of a security functional requirement, and the interface(s) that does not enforce any security functional requirements. The interfaces are described in terms of their: • purpose (general goal of the interface) • method of use (how the interface is to be used) • parameters (explicit inputs to and outputs from an interface that control the behaviour of that interface) • parameter descriptions (tells what the parameter is in some meaningful way) • error messages (identifies the condition that generated it, what the message is, and the meaning of any error codes) The development evidence also contains a tracing of the interfaces to the SFRs described in this ST.
AGD_OPE.1	The Administrative Guide provides the descriptions of the processes and procedures of how the administrative users of the TOE can securely administer the TOE using the interfaces that provide the features and functions detailed in the ST.
AGD_PRE.1	The Installation Guide describes the installation, generation and start-up procedures so that the users of the TOE can setup the components of the TOE into the evaluated configuration.
ALC_CMC.1 ALC_CMS.1	The CM[6] document(s) describes how the consumer (end-user) of the TOE can identify the evaluated TOE. The CM document(s) identifies the configuration items, how those configuration items are uniquely identified, and the adequacy of the procedures that are used to control and track changes that are made to the TOE. This includes details on what changes are tracked, how potential changes are incorporated, and the degree to which automation is used to reduce the scope for error.
ATE_IND.1	Cisco will provide the TOE for testing.
AVA_VAN.1	Cisco will provide the TOE for testing.

---

[6] CM – Configuration Management

6   TOE Summary Specification

6.1   TOE Security Functional Requirement Measures

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

Table 18 How TOE SFRs Measures

TOE SFRs	How the SFR is Met
FAU_GEN.1	The TOE generates an audit record whenever an audited event occurs. The types of events that cause audit records to be generated include start-up and shut-down 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 Table 14 above. Each of the events is specified in the audit record is in enough detail to identify the user for which the event is associated, when the event occurred, where the event occurred, the outcome of the event, and the type of event that occurred such as generating keys, including the type of key. Additionally, the start-up and shut-down 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 the required information. Additional information can be configured.
FAU_GEN.2	The TOE associates each auditable event 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.
FAU_STG_EXT.1	The TOE is a standalone device configured to export syslog records to a specified, external syslog server in real-time. The TOE protects communications with an external syslog server via IPsec. If the IPsec connection fails, the TOE will store audit records on the TOE when it discovers it can no longer communicate with its configured syslog server. When the connection is restored, the TOE will transmit the buffer contents to the syslog server. For audit records stored internally to the TOE the audit records are stored in a circular log file where the TOE overwrites the oldest audit records when the audit trail becomes full. The size of the logging files on the TOE is configurable by the Administrator with the minimum value being 4096 (default) to 2,147,483,647 bytes of available disk space Refer to the Common Criteria Operational User Guidance and Preparative Procedures for command description and usage information. Only Authorized Administrators can 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 FCS_CKM.2	The following table describes the key generation algorithms the TOE implements to generate asymmetric keys used for **device authentication**: *(table below)* With the exception to SSH, the keys are used to generate certificate signing requests (CSRs) in which the public key is associated with an X.509 certificate.

Scheme	Standard	Key Size/ NIST Curve	SFR	Service
RSA	FIPS PUB 186-4	2048 3072	FCS_SSHS_EXT.1	SSH Remote Administration
			FCS_IPSEC_EXT.1	Transmit generated audit data to an external IT entity Remote RADIUS authentication

TOE SFRs	How the SFR is Met
	The following table shows the key generation algorithms the TOE implements to generate asymmetric keys used for **key establishment**:

Scheme	Standard	Key Size/ NIST Curve	SFR	Service
FFC	NIST SP 800-56A Revision 3; RFC 3526	DH Group 14 (2048 bit)	FCS_IPSEC_EXT.1	Transmit generated audit data to an external IT entity

Remote RADIUS authentication |
| RSA | FIPS PUB 186-4 | 2048 | FCS_SSHS_EXT.1 | SSH Remote Administration |

The following table shows the methods the TOE implements for **key establishment**:

Scheme	Standard	SFR	Service
FFC	NIST SP 800-56A Revision 3; Section 3 of RFC 3526 for DH Group 14	FCS_IPSEC_EXT.1	Transmit generated audit data to an external IT entity

Remote RADIUS authentication |
| RSAES-PKCS1-v1_5 | Section 7.2 of RFC 3447 | FCS_SSHS_EXT.1 | SSH Remote Administration |

TOE SFRs	How the SFR is Met
FCS_CKM.4	The TOE meets all requirements specified in FIPS 140-2 for destruction of keys and Critical Security Parameters (CSPs) when no longer required for use. See section 7 below for additional details on key zeroization.
FCS_COP.1/DataEncryption	The TOE provides symmetric encryption and decryption capabilities using AES in CBC mode (128 and 256 bits) as described in ISO/IEC 18033-3 and ISO/IEC 10116. AES is implemented in the SSH and IPsec protocols. Refer to Table 4 above for the FIPS validated algorithm certificate numbers.
FCS_COP.1/SigGen	The TOE provides cryptographic signature services using a RSA Digital Signature Algorithm with key size of 2048 or 3072 as specified in FIPS PUB 186-4. Refer to Table 4 above for the FIPS validated algorithm certificate numbers.
FCS_COP.1/Hash
FCS_COP.1/KeyedHash | The TOE provides cryptographic hashing services using SHA-1, SHA-256 and SHA-512 as specified in ISO/IEC 10118-3:2004 (with key sizes and message digest sizes of 160, 256, and 512 bits respectively).

The TOE provides keyed-hashing message authentication services using HMAC-SHA-1 and HMAC-SHA-256 that operates on 512-bit blocks and HMAC-SHA-512 operating on 1024-bit blocks of data, with key sizes and message digest sizes of 160-bits, 256 bits and 512 bits respectively as specified in ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2".

For IKE Internet Security Association and Key Management Protocol (ISAKMP) hashing, Administrators configure the SHA and message digest to be used with remote IPsec endpoints. |

TOE SFRs	How the SFR is Met
	For IPsec Security Association (SA) authentication integrity options Administrators can select any of esp-sha-hmac (HMAC-SHA-1), esp-sha256-hmac (HMAC-SHA-256), or esp-sha512-hmac (HMAC_SHA-512) with message digest sizes of 160 and 256 and 512 bits respectively to be part of the IPsec SA transform-set to be used with remote IPsec endpoints. For SSH remote administration, Administrators configure hmac-sha2-256 and hmac-sha2-512 for MAC integrity. SHA-512 hashing is used for verification of software image integrity. Refer to Table 4 above for the FIPS validated algorithm certificate numbers.

FCS_COP.1(1)/KeyedHashCMAC Cryptographic Operation (AES-CMAC Keyed Hash Algorithm)
 	
FCS_COP.1(5) Cryptographic Operation (MACsec Data Encryption/Decryption)
	The TOE implements AES-CMAC keyed hash function for message authentication as described in NIST SP800-38B.
	
	The key length, hash function used, block size, message digest and output MAC length used are as follows:
	
	AES-CMAC 128 (hash function and key length)
	   Block Sizes: Full (block size)
	   Message digest size: 128 bits
	   Message Length: 0-256 bits (output MAC length)
	
	The TOE provides symmetric encryption and decryption capabilities using AES in AES Key Wrap and GCM mode (128 bits) as described in AES as specified in ISO/IEC 18033-3, AES Key Wrap as specified in NIST SP800-38F, GCM as specified in ISO/IEC 19772.
	
	AES is implemented in the MACsec protocol.
	
	Refer to Table 4 above for the FIPS validated algorithm certificate numbers.

FCS_IPSEC_EXT.1	The TSF implements IPsec to provide authentication and encryption services to prevent unauthorized viewing or modification of syslog and RADIUS authentication data as it travels over the external network. The TSF's implementation of the IPsec standard (in accordance with the RFCs noted in the SFR) uses the Encapsulating Security Payload (ESP) protocol to provide authentication and encryption supporting the following algorithms: ■ AES (AES-CBC-128 (RFC 3602) and AES-CBC-256 (RFC 3602) with a SHA-based HMAC (HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-512) to implement IKEv2 and the IPsec protocol ESP as defined in RFC 4303. The TOE supports both transport and tunnel mode for IPsec communications between the TOE and an external audit and authentication server. The administrator defines the traffic that needs to be protected between two IPsec peers by configuring access lists and applying these access lists to interfaces using crypto map sets. A crypto map set can contain multiple entries, each with a different access list. The crypto map entries are searched in a sequence--the router attempts to match the packet to the access list specified in that entry. When a packet matches a permit entry in a particular access list, and the corresponding crypto map entry is tagged connections are established, if necessary. If the crypto map entry is tagged as ipsec-isakmp, IPsec is triggered. If there is no SA that the IPsec can use to protect this traffic to the peer, IPsec uses IKE to negotiate with the re-mote 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. Once established, the set of SAs (outbound to the peer) is then applied to the triggering packet and to subsequent applicable packets as those packets exit the Controller. "Applicable" packets are packets that match the same access list criteria that the original packet matched. For example, all applicable packets could be encrypted be-fore being forwarded to the remote peer. The corresponding inbound SAs are used when processing the incoming traffic from that peer. Access lists associated with IPsec crypto map entries also represent the traffic that the Controller needs protected by IPsec. Inbound traffic is processed against crypto map entries. if an unprotected packet matches a permit entry in a particular access list associated with an IPsec crypto map entry, that packet is dropped because it was not sent as an IPsec-protected packet. The traffic matching the permit ACLs would

TOE SFRs	How the SFR is Met
	then flow through the IPsec tunnel and be classified as "PROTECTED". Traffic that does not match a permit ACL in the crypto map, but that is not disallowed by other ACLs on the interface is allowed to BYPASS the tunnel. Traffic that does not match a permit ACL and is also blocked by other non-crypto ACLs on the interface would be DISCARDED. Rules applied to an access control list can be applied to either inbound or outbound traffic. IPsec Internet Key Exchange, also called ISAKMP, is the negotiation protocol that lets two peers agree on how to build an IPsec Security Association (SA). The strength of the symmetric algorithm negotiated to protect the IKEv2 IKE_SA connection is greater than or equal to the strength of the symmetric algorithm negotiated to protect the IKEv2 CHILD_SA connection. The IKE protocols implement Peer Authentication using RSA X.509v3 certificates or pre-shared keys. IKE separates negotiation into two phases: IKEv2 SA and IKEv2 Child SA. The IKEv2 SA creates the first tunnel, which protects later ISAKMP negotiation messages. The key negotiated during the IKEv2 SA enables IKE peers to negotiate IKE v2 Child SA and establishes the IPsec SA to communicate securely. 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. The resulting potential strength of the symmetric key will be 128 or 256 bits of security depending on the algorithms negotiated between the two IPsec peers. As part of this negotiation, the TOE verifies that the negotiated IKE Child SA symmetric algorithm key strength is at most as large as the negotiated IKE SA key strength as configured on the TOE and peer via an explicit check. Each IKE negotiation begins by agreement of both peers on a common (shared) IKE policy. This policy states which security parameters will be used to protect subsequent IKE negotiations and mandates how the peers are authenticated. The Security Administrator can configure multiple, prioritized policies on each peer, each with a different combination of parameter values. However, at least one of these policies must contain exactly the same encryption, hash, authentication, and Diffie-Hellman parameter values as one of the policies on the remote peer. For each policy created, the Security Administrator assign's a unique priority (1 through 10,000, with 1 being the highest priority). When the IKE negotiation begins, IKE searches for an IKE policy that is the same on both peers. The peer that initiates the negotiation will send all its policies to the remote peer, and the remote peer will try to find a match. The remote peer looks for a match by comparing its own highest priority policy against the policies received from the other peer. The remote peer checks each of its policies in order of its priority (highest priority first) until a match is found. 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. When a packet is processed by the TOE and it determines it requires IPsec, it uses active SA settings or creates new SAs for initial connections with the IPsec peer. The TOE supports IKEv2 session establishment. The TOE supports configuration of session lifetimes for both IKEv2 SAs and IKEv2 Child SAs using the following the command "lifetime." The time values for IKEv2 SAs can be limited up to 24 hours and for IKEv2 Child SAs up to 8 hours. The IKEv2 Child SA lifetimes can also be configured by an Administrator based on number of bytes. The TOE supports Diffie-Hellman Group 14. The TSF generates the secret value 'x' used in the IKEv2 Diffie-Hellman key exchange ('x' in g^x mod p) using the NIST approved DRBG specified in FCS_RBG_EXT.1 and having possible length 224 bits. The TOE generates nonces used in IKEv2 exchanges, of at least 128 bits in size and at least half the output size of the negotiated pseudorandom function (PRF) hash. When a random number is needed for a nonce, the probability that a specific nonce value will be repeated during the life a specific IPsec SA is less than 1 in 2^{128}. The nonce is likewise generated using the AES-CTR DRBG. The TOE supports authentication of IPsec peers using pre-shared keys and RSA X.509 certificates. During IKE establishment, IPsec peers authenticate each other by creating and exchanging a hash value that includes the pre-shared key. The TOE will compare the received hash value to its computed hash and determine if it matches. If it does, pre-shared key authentication is successful; otherwise pre-shared key authentication fails.

TOE SFRs	How the SFR is Met
	For peer authentication using RSA certificates, the TOE validates the presented identifier provided supporting the following fields and types: SAN: IP address, SAN: Fully Qualified Domain Name (FQDN). Certificate maps provide the ability for a certificate to be matched with a given set of criteria. The Administrator is instructed in the CC Configuration Guide to specify one or more certificate fields together with their matching criteria and the value to match. In the evaluated configuration, the field name must specify the SAN (alt-subject-name) field. Match criteria should be "eq" for equal. SAN example: alt-subject-name eq <peer.cisco.com> The TOE will reject the IKE connection in any of these situations: 1) If the data ID Payload for any of those ID Types does not match the peer's certificate exactly; 2) If an ID Payload is not provided by the peer; 3) If multiple ID Types are provided in the ID Payload. When using pre-shared key the TOE will reference the match identity setting as configured in its own admin-defined settings to match the peer's IP address to the corresponding reference identifier.
FCS_MACSEC_EXT.1	The TOE implements MACsec in compliance with Institute of Electrical and Electronics Engineers (IEEE) Standard 802.1AE-2006. The MACsec connections maintain confidentiality of transmitted data and takes measures against frames transmitted or modified by unauthorized devices. The Secure Channel Identifier (SCI) is composed of a globally unique 48-bit Message Authentication Code (MAC) Address and the Secure System Address (port). The SCI is part of the SecTAG if the Secure Channel (SC) bit is set and will be at the end of the tag. Any MAC Protocol Data Units (MPDUs) during a given session that contain an SCI other than the one used to establish that session is rejected. Only Extensible Authentication Protocol over LAN (EAPOL) (Physical Address Extension (PAE) EtherType 88-8E), MACsec frames (EtherType 88-E5), and MAC control frames (EtherType 88-08) are permitted. All others are rejected.
FCS_MACSEC_EXT.2	The TOE implements the MACsec requirement for integrity protection with the confidentiality offsets of 0, 30 and 50 using the '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) of 16-bytes derived with the SAK 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 Initialization Vector (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 a Random Number Generator (RNG) each time an SAK is generated. 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 key size is 32-bit hexadecimal in length for AES 128-bit CMAC mode encryption.
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/IEC 18033-3, AES in CMAC mode as specified in NIST SP800-38B, and GCM as specified in ISO/IEC 19772.

TOE SFRs	How the SFR is Met
	The 'Key-chain macsec lifetime' configuration command is used to specify the lifetime for CAKs. The 'MACSEC Key-chain key' command is used to specify the length of the CKN. The CKN can be set between 1 and 32 octets.
FCS_MKA_EXT.1	The TOE implements the MKA Protocol in accordance with IEEE 802.1X-2010 and 802.1Xbx-2014. The data delay protection is enabled for MKA as a protection guard against an attack on the configuration protocols that MACsec is designed to protect by alternately delaying and delivering their MPDUs. The "Delay Protection" does not operate if MKA operation is suspended. An MKA Lifetime Timeout limit of 6.0 seconds and Hello Timeout limit of 2.0 seconds is enforced by the TOE. The TOE discards MACsec Key Agreement Protocol Data Units (MKPDUs) that do not satisfy the requirements listed under FCS_MKA_EXT.1.8 in Section 5.2.2.15. All valid MKPDUs that meet the requirements as defined under FCS_MKA_EXT.1.8 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 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 CAK that is used for ICV validation by the MKA protocol.

FCS_SSHS_EXT.1	The TOE implementation of SSHv2 supports the following:
	The TSF implements SSHv2 conformant to RFCs 4251, 4252, 4253, 4254, 6668, 8308 section 3, and 8332 to provide a secure command line interface for remote administration. The TOE uses rsa-sha2-512 and rsa-sha2-256 for host key authentication and uses ssh-rsa for client or user password-based authentication.
	
	SSHv2 connections will be dropped if the TOE receives a packet larger than 33,038 bytes. Large packets are detected by the SSHv2 implementation and dropped internal to the SSH process.
	
	The TSF's SSH transport implementation supports the following encryption algorithms:
	
	    ■ aes128-cbc
	
	    ■ aes256-cbc
	
	All connection attempts from remote SSH clients requesting any other encryption algorithm is denied.
	
	The TSF's SSH transport implementation supports the following MAC algorithms:
	
	    ■ hmac-sha2-256
	
	    ■ hmac-sha2-512
	
	All connection attempts from remote SSH clients requesting any other MAC algorithm is denied.
	
	The TSF's SSH transport implementation supports the following public-key algorithms for Hostkey authentication:
	
	    ■ rsa-sha2-256
	
	    ■ rsa-sha2-512
	
	The TSF's SSH transport implementation supports the following public-key algorithms for Client Authentication:
	
	    ■ ssh-rsa

TOE SFRs	How the SFR is Met
	When the SSH client presents a public key, the TSF verifies it matches the one configured for the Administrator account. If the presented public key does not match the one configured for the Administrator account, access is denied. The TSF's SSH key exchange implementation supports the following key exchange algorithm: ■ diffie-hellman-group14-sha1 The TSF's SSH implementation will perform a rekey after no longer than one hour or more than one gigabyte of data has been transmitted with the same session key. Both thresholds are checked. Rekeying is performed upon reaching whichever threshold is met first. The Administrator can configure lower rekey values if desired. The minimum time value is 10 minutes. The minimum volume value is 100 kilobytes.
FCS_RBG_EXT.1	The TOE implements a NIST-approved AES-CTR DRBG, as specified in ISO/IEC 18031:2011 seeded by an entropy source that accumulates entropy from a TSF-platform based noise source. The DRBG 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.
FIA_AFL.1	To block password-based brute force attacks, the TOE uses an internal AAA function to detect and track failed login attempts. When an account attempting to log into an administrative interface reaches the set maximum number of failed authentication attempts, the account will not be granted access until the time period has elapsed. The TOE provides the Administrator the ability to specify the maximum number of unsuccessful authentication attempts before an offending account will be blocked. The TOE also provides the ability to specify the time period to block offending accounts. To avoid a potential situation where password failures made by Administrators leads to no Administrator access until the defined blocking time period has elapsed, the CC Configuration Guide instructs the Administrator to configure the TOE for SSH public key authentication which is not subjected to password-based brute force attacks. During the block out period, the TOE provides the ability for the Administrator account to login remotely using SSH public key authentication.
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: "!", "@", "#", "$", "%", "^", "&", "*", "(", ")" and other special characters listed in table 15. Minimum password length is settable by the Authorized Administrator, and can be configured for minimum password lengths of 1 and maximum of 127 characters. A minimum password length of 8 is recommended.
FIA_PSK_EXT.1	Through the implementation of the CLI, the TOE supports IKEv2 pre-shared keys for authentication of IPsec tunnels. Preshared keys can be entered as American Standard Code for Information Interchange (ASCII) character strings, or HEX values. The TOE supports keys that are from 1 character in length up to 127 bytes in length and composed of any combination of upper- and lower-case letters, numbers, and special characters (that include: "!", "@", "#", "$", "%", "^", "&", "*", "(", and ")". 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 or 64 characters in length.
FIA_UIA_EXT.1 FIA_UAU_EXT.2	The TOE requires all users to be successfully identified and authenticated before allowing any TSF mediated actions to be performed. Prior to being granted access, a login warning banner is displayed. Administrative access to the TOE is facilitated through a local password-based authentication and SSH public key authentication mechanisms on the TOE through which all Administrator actions are mediated. Once a potential (unauthenticated) administrative user attempts to access the TOE through an interactive administrative interface, the TOE prompts the user for a user name and password or SSH public key authentication. No access is allowed to the administrative functionality of the TOE until the administrator is successfully identified and authenticated. The TOE provides a local password-based authentication mechanism as well as RADIUS AAA server for remote authentication.

TOE SFRs	How the SFR is Met
	The Administrator authentication policies include authentication to the local user database or redirection to a remote authentication server. Interfaces can be configured to try one or more remote authentication servers, and then fail back to the local user database if the remote authentication servers are inaccessible. The process for authentication is the same for administrative access whether administration is occurring via a directly connected console or remotely via SSHv2 secured connection. At initial login, the administrative user is prompted to provide a username. After the user provides the username, the user is prompted to provide the administrative password associated with the user account. The TOE then either grants administrative access (if the combination of username and password is correct) or indicates that the login was unsuccessful. The TOE does not provide a reason for failure in the cases of a login failure.
FIA_UAU.7	When a user enters their password at the local console, the TOE does not echo any characters as the password is entered. For remote session authentication, the TOE does not echo any characters as they are entered.
FIA_X509_EXT.1/Rev	The TOE uses X.509v3 certificates to support authentication for IPsec connections. The TSF determines the validity of certificates by ensuring that the certificate and the certificate path are valid in accordance with RFC 5280. The certificate path is validated by ensuring that all the CA certificates have the basicConstraints extension and the CA flag is set to TRUE and the certificate path must terminate with a trusted CA certificate. CRL revocation checking is supported by the TOE. Revocation checking is performed on the leaf and intermediate certificate(s) when authenticating a certificate chain provided by the remote peer. There are no functional differences if a full certificate chain or only a leaf certificate is presented.
FIA_X509_EXT.2	The TOE determines which certificate to use based upon the trustpoint configured. The instructions for configuring trustpoints is provided in CC Configuration Guide. In the event that a network connection cannot be established to verify the revocation status of certificate for an external peer the connection will be rejected.
FIA_X509_EXT.3	A Certificate Request Message can be generated as specified by RFC 2986 and provide the following information in the request – Common Name(CN), Organization(O), Organizational Unit(OU), and Country(C). The TOE can validate the chain of certificates from the Root CA when the CA Certificate Response is received.
FMT_MOF.1/ManualUpdate FMT_MTD.1/CoreData FMT_MTD.1/CryptoKeys	The TOE provides the ability for Security Administrators to access TOE data, such as audit data, configuration data, security attributes, routing tables, and session thresholds and to perform manual updates to the TOE. Only Security Administrators can access the TOE's trust store. Each of the predefined and administratively configured roles has create (set), query, modify, or delete access to the TOE data, though with some privilege levels, the access is limited. The TOE performs role-based authorization, using TOE platform authorization mechanisms, to grant access to the privileged and semi-privileged roles. For the purposes of this evaluation, the privileged level is equivalent to full administrative access to the CLI, which is the default access for IOS-XE privilege level 15; and the semi-privileged level 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 customizable. The term "Authorized Administrator" is used in this ST to refer to any user that 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 semi-privileged Administrators with only a subset of privileges may also manage and modify TOE data based on the privileges assigned. The TOE provides the ability for Authorized Administrators to access TOE data, such as audit data, configuration data, security attributes, session thresholds, cryptographic keys, and updates. Each of the predefined and administratively configured privilege levels has a set of permissions that will grant access to the TOE data, though with some privilege levels, the access is limited. The TOE does not provide automatic updates to the software version running on the TOE. The Authorized Administrator can query the software version running on the TOE and can initiate updates to (replacements of) software images. When software updates are made available by Cisco, the Authorized Administrators can obtain, verify the integrity of, and install those updates. The Authorized Administrator generates RSA key pairs to be used in the IKE and SSH protocols. Zeroization of these keys is provided in Table 19 below.

TOE SFRs	How the SFR is Met
	Prior to authentication the TOE may be configured by the Administrator to display a customized login banner, which describes restrictions of use, legal agreements, or any other appropriate information to which users consent by accessing the TOE. No administrative functionality is available prior to administrative login. TOE Administrators can control (generate/delete) the following keys, IKE RSA Key Pairs and SSH RSA Key Pairs by following the instruction in the AGD.
FMT_SMF.1	The TOE provides all 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 Authorized Administrator can perform all management functions by accessing the TOE directly via connected console cable or remote administration via SSHv2 secure connection. The specific management capabilities available from the TOE include: • Local and remote administration of the TOE and the services provided by the TOE via the TOE CLI, as described above • The ability to manage the warning banner message and content which allows the Authorized Administrator the ability to define warning banner that is displayed prior to establishing a session (note this applies to the interactive (human) users, e.g., administrative users • The ability to set and modify the time limits of session inactivity • The ability to configure the number of failed Administrator logon attempts that will cause the account to be locked until a specified time period has elapsed. • The ability to update the IOS-XE software. The validity of the image is provided using a digital signature prior to installing the update • The ability to modify the behavior of the transmission of audit data to an external IT entity • The ability to manage cryptographic keys • The ability to manage the cryptographic functionality which allows the Authorized Administrator the ability to identify and configure the algorithms used to provide protection of the data, such as generating the RSA keys to enable SSHv2 • The ability to configure the IPsec functionality which supports the secure connections to the audit server and the remote authentication server • Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchor • Ability to configure thresholds for SSH rekeying • The ability to manage the trusted public keys database • The ability to manage the Key Server and associated MKA participants • The ability to generate a PSK and install in the CAK cache • The ability to initiate the generation of a new CAK from the Key Server • The ability to specify the lifetime of a CAK and to enable, disable or delete a PSK in the CAK cache of a device • The ability to configure and set the time clock • The ability to configure the reference identifiers for peers, which can be SAN: IP Address or SAN: FQDN.
FMT_SMR.2	The TOE maintains privileged and semi-privileged Administrator roles. The TOE performs role-based authorization, using TOE platform authorization mechanisms, to grant access to TOE functions. 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 (PL) 15. Semi-privileged roles are assigned a PL of 0 – 14. PL 0 and 1 are defined by default and are customizable, while PL 2-14 are undefined by default and are also customizable. Note: Levels 0 – 14 are a subset of PL 15 and the levels are not hierarchical. The term "Authorized Administrator" is used in this ST to refer to any user which has been assigned to a privilege level that is permitted to perform the relevant action; therefore, has the appropriate privileges to perform the requested functions. The privilege level determines the functions the user can perform, hence the Authorized Administrator with the appropriate privileges. The TOE can and shall be configured to authenticate all access to the command line interface using a username and password. The TOE supports both local administration via a directly connected console cable and remote administration via SSHv2 secure connection.

TOE SFRs	How the SFR is Met
FPT_CAK_EXT.1	A CAK is specified in the configuration file by the Administrator using a bit-based (hex) format. Only the Administrator may view the configuration file.
FPT_FLS.1.1(2)/SelfTest	Whenever a failure occurs (power-on self-tests, integrity check of the TSF executable image and/or the noise source health-tests) 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. If the failures persist, the TOE will continue to reload in an attempt to correct the failure. This functionally prevents any failure from causing an unauthorized information flow. There are no failures that circumvent this protection. If the rebooting continues, the Authorized Administrator must contact Cisco Technical Assistance Center (TAC).
FPT_RPL.1	Replayed data is discarded by the TOE and the attempt to replay data is logged. MKPDUs are replay protected in the TOE. The MKA frames are guarded against replay, such that if a MKPDU contains a duplicate Member Number (MN) and not the most current MN, then this MKPDU will be dropped and not processed further. In addition, the attempt to replay data is logged.
FPT_SKP_EXT.1	The TOE is designed specifically to not disclose any keys stored in the TOE. The TOE stores all private keys in a secure directory that cannot be viewed or accessed, even by the Administrator. The TOE stores symmetric keys only in volatile memory. Pre-shared keys may be specified in the configuration file by the Administrator using a bit-based (hex) format. Only the Administrator may view the configuration file.
FPT_APW_EXT.1	The TOE is designed specifically to not disclose any passwords stored in the TOE. All passwords are stored using a SHA-2 hash. 'Show' commands display only the hashed password. The CC Configuration Guide instructs the Administrator to use the algorithm-type scrypt sub-command when passwords are created or updated. The scrypt is password type 9 and uses a SHA-2 hash. ..
FPT_STM_EXT.1	The TSF implements a clock function to provide a source of date and time. The clock function is reliant on the system clock provided by the underlying hardware. All Switch models have a real-time clock (RTC) with battery to maintain time across reboots and power loss. The TOE relies upon date and time information for the following security functions: ■ To monitor local and remote interactive administrative sessions for inactivity (FTA_SSL_EXT.1, FTA_SSL.3); ■ Validating X.509 certificates to determine if a certificate has expired (FIA_X509_EXT.1/Rev), ■ To determine when SSH session keys have expired and to initiate a rekey (FCS_SSHS_EXT.1); ■ To determine when IKEv2 SA lifetimes have expired and to initiate a rekey (FCS_IPSEC_EXT.1); ■ To determine when IPsec Child SA lifetimes have expired and to initiate a rekey (FCS_IPSEC_EXT.1); ■ To provide accurate timestamps in audit records (FAU_GEN.1.2).
FPT_TUD_EXT.1	An Authorized Administrator can query the software version running on the TOE and can initiate updates to (replacements of) 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 the updates. The updates can be downloaded from https://software.cisco.com/software/csws/ws/platform/home?locale=en_US# A digital signature is used to verify software files (to ensure they have not been modified from the originals distributed by Cisco) before installing. If the integrity check fails, the software is not loaded and the system reboots to attempt the test again. If the test continues to fail, the Authorized Administrator must contact Cisco. If the integrity check is successful, the software is loaded and the device continues with the bootup process. To verify the digital signature prior to installation, the "show software authenticity file" command displays 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 TAC.

TOE SFRs	How the SFR is Met
	Once the integrity check is complete, the power-on self-tests are executed. If the power-on self-tests are successful, the TOE continues to load into an operational state. If a power-on self-test fails, the TOE automatically reboots to attempt to clear the error state. The TOE will continue to reboot until the error is cleared and the device is operational. If the error persists, the Authorized Administrator must contact Cisco.
FPT_TST_EXT.1	The TOE runs a suite of self-tests during initial start-up to verify correct operation of the cryptographic module. 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 any of the tests fail, the system halts and a message is displayed to the local console. These tests include: ***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. If the encrypted texts match, the test passes; otherwise, the test fails. The decrypt test is just the opposite. In this test a known key is used to decrypt a known encrypted value. The resulting plaintext value is compared to a known plaintext value. If the decrypted texts match, the test passes; otherwise, the test fails. ***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. If the encrypted values, the test passes; otherwise, the test fails. The encrypted data is then decrypted using the private key. This value is compared to the original plaintext value. If the decrypted values match, the test passes; otherwise, the test fails. ***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. If the random bits match, the test passes; otherwise, the test fails. ***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. If the MAC values match, the test passes; otherwise, the test fails. ***Software Integrity Test:*** The Software Integrity Test is run automatically whenever the module is loaded and confirms the module has maintained its integrity. ***SHA-1/256/512 Known Answer Test:*** For each of the values listed, the SHA implementation is fed known data and a key. These values are used to generate a hash. This hash is compared to a known value. If the hash values match, the test passes; otherwise, the test fails. If any component reports failure for the POST, the system crashes. Appropriate information is displayed on the screen and saved in the crashinfo file. All ports are blocked during the POST. If all components pass the POST, the system is placed in FIPS PASS state and ports can forward data traffic. If an error occurs during the self-test, a SELF_TEST_FAILURE system log is generated. Example Error Message: `%CRYPTO-0-SELF_TEST_FAILURE: Crypto algorithms self-test failed (SHA hashing)` 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 behaviour will be identified by the failure of a self-test.
FTA_SSL_EXT.1 FTA_SSL.3	The Administrator can configure maximum inactivity times individually for both local and remote administrative sessions. If either the local or remote administrative sessions are inactive for a configured period of time, the session will be terminated and will require re-authentication. The local interactive session terminates and does not lock. An Authorized Administrator can configure maximum inactivity times individually for both local and remote administrative sessions using the "`exec-timeout`" command applied to the console and virtual terminal (vty) lines. The allowable inactivity timeout range is from is <0-35791> minutes.

TOE SFRs	How the SFR is Met
	The configuration of the vty lines sets the configuration for the remote console access. The line console settings are not immediately activated for the current session. The current line console session must be exited. When the user logs back in, the inactivity timer will be activated for the new session..
FTA_SSL.4	An Authorized Administrator can exit out of both local and remote administrative sessions by issuing the 'exit' command.
FTA_TAB.1	The Administrator can configure an access banner that describes restrictions of use, legal agreements, or any other appropriate information to which users consent by accessing the TOE. The banner will display on the local console port and SSH interfaces prior to allowing any administrative access.
FTP_ITC.1	The TOE uses secure protocols to provide trusted communications between itself and authorized IT entities as specified in the table below: <table><tr><th>IT Entity</th><th>Secure Communication Mechanism/ Protocol</th><th>Non-TSF Endpoint Identification</th></tr><tr><td>Syslog Server</td><td>IPsec</td><td>X.509 Certificate</td></tr><tr><td>AAA/RADIUS Server</td><td>IPsec</td><td>X.509 Certificate</td></tr><tr><td>MACsec Peer</td><td>MACsec</td><td>Pre-Shared Key</td></tr></table>
FTP_TRP.1/Admin	All remote administrative communications take place over a secure encrypted SSHv2 session. The SSHv2 session is encrypted using AES encryption. The remote users (Authorized Administrators) 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. As described below in the table, the TOE zeroize all secrets, keys, and associated values when they are no longer required. The process in which the TOE zeroizes, meets FIPS 140 validation.

Table 19 TOE Key Zeroization

Name	Description	Zeroization
DH Shared Secret	The value is zeroized after it has been given back to the consuming operation. The value is overwritten by 0's. This key is stored in Dynamic Random-Access Memory (SDRAM).	Automatically after completion of DH exchange. Overwritten with: 0x00
DH 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 an IKE intermittent value used to create skeyid_d. This information is stored in SDRAM.	Automatically after IKE session terminated. Overwritten with: 0x00
skeyid_d	This is an IKE intermittent value used to derive keying data for IPsec. This information is stored in SDRAM.	Automatically after IKE session terminated. Overwritten with: 0x00
IKE session encrypt key	This the key IPsec key used for encrypting the traffic in an IPsec connection. This key is stored in SDRAM.	Automatically after IKE session terminated. Overwritten with: 0x00
IKE session authentication key	This the key IPsec key used for authenticating the traffic in an IPsec connection. This key is stored in SDRAM.	Automatically after IKE session terminated. Overwritten with: 0x00
ISAKMP preshared	This is the configured pre-shared key for ISAKMP negotiation. This key is stored in NVRAM.	Zeroized using the following command: # no crypto isakmp key[7] Overwritten with: 0x0d
IKE RSA Private Key	The RSA private-public key pair is created by the device itself using the key generation CLI described below. The device's public key must be added 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 to enrol with the CA server to generate the device certificate. In the IKE authentication step, the device's certificate is first sent to another device so that it can be authenticated. The other device verifies the certificate is signed by CA's signing key, and then the device sends a random secret encrypted by the device's public key in the valid device certificate. Thus, establishing the trusted connection since 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[8] Overwritten with: 0x0d

---

[7] Using this command will zeroize all isakmp keys.
[8] Using this command will zeroize all RSA keys.

Name	Description	Zeroization
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
MACsec SAK	The SAK is used to secure the control plane traffic. This key is stored in internal ASIC register.	Automatically when MACsec session terminated. The value is zeroized by overwriting with another key or freed when the session expires.
MACsec CAK	The CAK secures the control plane traffic. This key is stored in internal ASIC register.	Automatically when MACsec session terminated. The value is zeroized by overwriting with another key or freed when the session expires.
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 (SCA). This key is stored in internal ASIC register.	Automatically when MACsec session terminated. The value is zeroized by overwriting with another key or freed when the session expires.
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 terminated. The value is zeroized by overwriting with another key or freed when the session expires.
RADIUS secret	Shared secret used as part of the RADIUS authentication method. The password is stored in NVRAM.	Zeroized using the following command: # no radius-server key[9] Overwritten with: 0x0d
SSH Private Key	Once the function has completed the operations requiring the RSA key object, the module overwrites the entire object (no matter its contents). This key is stored in NVRAM.	Zeroized using the following command: # crypto key zeroize rsa [10] Overwritten with: 0x00
SSH Session Key	Once the function has completed the operations requiring the RSA key object, the module overwrites the entire object (no matter its contents). This key is stored in SDRAM.	Automatically when the SSH session is terminated. Overwritten with: 0x00
User Password	This is a variable 1-127 character password that is used to authenticate local users. The password is stored in NVRAM.	Zeroized by overwriting with a new password
Enable Password (if used)	This is a variable 1-127 character password that is used to authenticate local users at a higher privilege level. The password is stored in NVRAM.	Zeroized by overwriting with a new password

---

[9] Using this command will zeroize all radius-server keys.
[10] Using this command will zeroize all RSA keys

Name	Description	Zeroization
RNG Seed	This seed is for the RNG. The seed is stored in SDRAM.	Zeroized upon power cycle of the device
RNG Seed Key	This is the seed key for the RNG. The seed key is stored in SDRAM.	Zeroized upon power cycle of the device

8   Annex B: NIAP Technical Decisions

This ST applies the following NIAP Technical Decisions:

Table 20 NIAP Technical Decisions

TD Identifier	TD Name	Protection Profiles	References	Publication Date	Applicable?
TD0738	NIT Technical Decision for Link to Allowed-With List	CPP_ND_V2.2E	Chapter 2	2023.05.19	Yes
TD0670	NIT Technical Decision for Mutual and Non-Mutual Auth TLSC Testing	CPP_ND_V2.2E	ND SD2.2, FCS_TLSC_EXT.2.1	2022.09.16	No, TLS Client not claimed
TD0652	MACsec CAK Lifetime in FMT_SMF.1	PP_NDCPP_MACSEC _EP_V1.2	FMT_SMF.1	2022.08.31	Yes
TD0639	NIT Technical Decision for Clarification for NTP MAC Keys	CPP_ND_V2.2E	FCS_NTP_EXT.1.2, FAU_GEN.1, FCS_CKM.4, FPT_SKP_EXT.1	2022.08.26	No, NTP not claimed
TD0638	NIT Technical Decision for Key Pair Generation for Authentication	CPP_ND_V2.2E	NDSDv2.2, FCS_CKM.1	2022.08.05	Yes
TD0636	NIT Technical Decision for Clarification of Public Key User Authentication for SSH	CPP_ND_V2.2E	NDSD2.2, FCS_SSHC_EXT.1	2022.03.21	No, SSH Client not claimed.
TD0635	NIT Technical Decision for TLS Server and Key Agreement Parameters	CPP_ND_V2.2E	FCS_TLSS_EXT.1.3, NDSD v2.2	2022.03.21	No, TLS Server not claimed.
TD0634	NIT Technical Decision for Clarification required for testing IPv6	CPP_ND_V2.2E	FCS_DTLSC_EXT.1.2, FCS_TLSC_EXT.1.2, ND SD v2.2	2022.03.21	No, [D]TLS Client not claimed.
TD0633	NIT Technical Decision for IPsec IKE/SA Lifetimes Tolerance	CPP_ND_V2.2E	NDSD2.2, FCS_IPSEC_EXT.1.7, FCS_IPSEC_EXT.1.8	2022.03.21	Yes
TD0632	NIT Technical Decision for Consistency with Time Data for vNDs	CPP_ND_V2.2E	NDSD2.2, FPT_STM_EXT.1.2	2022.03.21	Yes
TD0631	NIT Technical Decision for Clarification of public key authentication for SSH Server	CPP_ND_V2.2E	NDSDv2.2, FCS_SSHS_EXT.1, FMT_SMF.1	2022.03.21	Yes
TD0618	MACsec Key Agreement and conditional support for group CAK	PP_NDCPP_MACSEC _EP_V1.2	FCS_MKA_EXT.1.2, FCS_MKA_EXT.1.5, FCS_MKA.1.8	2022.02.07	Yes
TD0592	NIT Technical Decision for Local Storage of Audit Records	CPP_ND_V2.2E	FAU_STG	2021.05.21	Yes
TD0591	NIT Technical Decision for Virtual TOEs and hypervisors	CPP_ND_V2.2E	A.LIMITED_FUNCTIONA LITY, ACRONYMS	2021.05.21	No, the evaluation does not include a virtual TOE or hypervisor

TD Identifier	TD Name	Protection Profiles	References	Publication Date	Applicable?
TD0581	NIT Technical Decision for Elliptic curve-based key establishment and NIST SP 800-56Arev3	CPP_ND_V2.2E	FCS_CKM.2	2021.04.09	Yes
TD0580	NIT Technical Decision for clarification about use of DH14 in NDcPPv2.2e	CPP_ND_V2.2E	FCS_CKM.1.1, FCS_CKM.2.1	2021.04.09	Yes
TD0572	NiT Technical Decision for Restricting FTP_ITC.1 to only IP address identifiers	CPP_ND_V2.1, CPP_ND_V2.2E	FTP_ITC.1	2021.01.29	Yes
TD0571	NiT Technical Decision for Guidance on how to handle FIA_AFL.1	CPP_ND_V2.1, CPP_ND_V2.2E	FIA_UAU.1, FIA_PMG_EXT.1	2021.01.29	Yes
TD0570	NiT Technical Decision for Clarification about FIA_AFL.1	CPP_ND_V2.1, CPP_ND_V2.2E	FIA_AFL.1	2021.01.29	Yes
TD0569	NIT Technical Decision for Session ID Usage Conflict in FCS_DTLSS_EXT.1.7	CPP_ND_V2.2E	ND SD v2.2, FCS_DTLSS_EXT.1.7, FCS_TLSS_EXT.1.4	2021.01.28	No, SFR not claimed
TD0564	NiT Technical Decision for Vulnerability Analysis Search Criteria	CPP_ND_V2.2E	NDSDv2.2, AVA_VAN.1	2021.01.28	Yes
TD0563	NiT Technical Decision for Clarification of audit date information	CPP_ND_V2.2E	NDcPPv2.2e, FAU_GEN.1.2	2021.01.28	Yes
TD0556	NIT Technical Decision for RFC 5077 question	CPP_ND_V2.2E	NDSDv2.2, FCS_TLSS_EXT.1.4, Test 3	2020.11.06	No, SFR not claimed
TD0555	NIT Technical Decision for RFC Reference incorrect in TLSS Test	CPP_ND_V2.2E	NDSDv2.2, FCS_TLSS_EXT.1.4, Test 3	2020.11.06	No, SFR not claimed
TD0553	FCS_MACSEC_EXT.1.4 and MAC control frames	PP_NDCPP_MACSEC_EP_V1.2	FCS_MACSEC_EXT.1.4	2020.12.18	Yes
TD0547	NIT Technical Decision for Clarification on developer disclosure of AVA_VAN	CPP_ND_V2.1, CPP_ND_V2.2E	ND SDv2.1, ND SDv2.2, AVA_VAN.1	2020.10.15	Yes
TD0546	NIT Technical Decision for DTLS - clarification of Application Note 63	CPP_ND_V2.2E	FCS_DTLSC_EXT.1.1	2020.10.15	No, SFR not claimed
TD0537	The NIT has issued a technical decision for Incorrect reference to FCS_TLSC_EXT.2.3	CPP_ND_V2.2E	FIA_X509_EXT.2.2	2020.07.13	Yes
TD0536	The NIT has issued a technical decision for Update Verification Inconsistency	CPP_ND_V2.1, CPP_ND_V2.2E	AGD_OPE.1, ND SDv2.1, ND SDv2.2	2020.07.13	Yes

TD Identifier	TD Name	Protection Profiles	References	Publication Date	Applicable?
TD0528	The NIT has issued a technical decision for Missing EAs for FCS_NTP_EXT.1.4	CPP_ND_V2.1, CPP_ND_V2.2E	FCS_NTP_EXT.1.4, ND SD v2.1, ND SD v2.2	2020.07.13	No, SFR not claimed
TD0527	Updates to Certificate Revocation Testing (FIA_X509_EXT.1)	CPP_ND_V2.2E	FIA_X509_EXT.1/REV, FIA_X509_EXT.1/ITT	2020.07.01	Yes
TD0509	Correction to MACsec Audit	PP_NDCPP_MACSEC_EP_V1.2	FAU_GEN.1	2020.03.02	Yes
TD0487	Correction to Typo in FCS_MACSEC_EXT.4	PP_NDCPP_MACSEC_EP_V1.2	FCS_MACSEC_EXT.4.4	2020.01.02	Yes
TD0466	Selectable Key Sizes for AES Data Encryption/Decryption	PP_NDCPP_MACSEC_EP_V1.2	FCS_COP.1.1	2019.11.15	Yes
TD0273	Rekey after CAK expiration	PP_NDCPP_MACSEC_EP_V1.2	FCS_MACSEC_EXT.4	2017.12.20	Yes
TD0190	FPT_FLS.1(2)/SelfTest Failure with Preservation of Secure State and Modular Network Devices	PP_NDCPP_MACSEC_EP_V1.2	FPT_FLS.1(2)/SelfTest	2017.04.11	Yes
TD0135	SNMP in NDcPP MACsec EP v1.2	PP_NDCPP_MACSEC_EP_V1.2	FMT_SNMP_EXT.1.1, FCS_SNMP_EXT.1.1	2017.04.11	No, SFR not claimed

9   Annex C: Acronyms

Table 21 below provides a list of acronyms and abbreviations that are common and may be used in this Security Target.

Table 21 Acronyms

Acronyms / Abbreviations	Definition
AAA	Administration, Authorization, and Accounting
AC	Alternating Current
ACL (acl)	Access Control Lists
AES	Advanced Encryption Standard
AGD	Guidance Document
APT	Adaptive Proportion Test
ASCII	American Standard Code for Information Interchange
ASIC	Application Specific Integrated Circuit
CA	Connectivity Association
CAK	(Secure) Connectivity Association Key
CAVP	Cryptographic Algorithm Validation Program
CBC	Cipher Block Chaining
CC	Common Criteria for Information Technology Security Evaluation
CDP	CRL Distribution Point
CEM	Common Evaluation Methodology for Information Technology Security
CKN	Secure Connectivity Association Key Name
CLI	Command Line Interface
CM	Configuration Management
CMAC	Cipher Based Message Authentication Code
CPU	Central Processing Unit
CRL	Certificate Revocation List
CS	Certificate Server
CSP	Critical Security Parameter
CSR	Certificate Signing Request
CTR	Counter
CVL	Component Validation List
DH	Diffie-Hellman
DHCP	Dynamic Host Configuration Protocol
DM	Division Multiplexing
DN	Distinguished Name
SDRAM	Synchronous Dynamic Random-Access Memory
DRBG	Deterministic Random Bit Generator
DW	Dense Wavelength
EAL	Evaluation Assurance Level
EAP	Extensible Authentication Protocol
EAP-TLS	EAP Transport Layer Security
EAPOL	EAP over LANs
EEPROM	Electronically Erasable Programmable Read-Only Memory
EHWIC	Ethernet High-Speed WIC
ESP	Encapsulating Security Payload
FFC	Finite Field Cryptography
FQDN	Fully Qualified Domain Name
FRU	Field Replaceable Unit
GB	Giga Byte
GCM	Galois Counter Mode
GE	Gigabit Ethernet port
GUI	Graphical User Interface
HMAC	Hash-based Message Authentication Code
HTTP	Hypertext Transfer Protocol
HTTPS	HTTP Secure
IC2M	IOS Common Cryptographic Module
ICK	Integrity Check Key
ICMP	Internet Control Message Protocol
ICV	Integrity Check Value

Acronyms / Abbreviations	Definition
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
IFS	IOS-XE File System
IGMP	Internet Group Management Protocol
IKE	Internet Key Exchange
IOS	Internetworking Operating System
IP	Internet Protocol
IPsec	IP Security
ISAKMP	Internet Security Association and Key Management Protocol
ISDN	Integrated Services Digital Network
ISO	International Organization of Standardization
IT	Information Technology
KDF	Key Derivation Function
KEK	Key Encryption Key
KAS	Key Agreement Scheme
KAS-SSC	KAS-Shared Secret Computation
KW	Key Wrap
LC	Lucent Connector
MAC	Media Access Control
MACsec	MAC Security
MKA	MACsec Key Agreement protocol
MKPDU	MACsec Key Agreement Protocol Data Unit
MN	Member Number
MPDU	MAC Protocol Data Unit
MSAP	MAC Service Access Point
MSC	MACsec Controller
MSDU	MAC Service Data Unit
MSK	Master Session Key
NDcPP	collaborative Network Device Protection Profile
NIST	National Institute of Standards and Technology
NVRAM	Non-Volatile Random-Access Memory
OCSP	Online Certificate Status Protocol
OS	Operating System
OSI	Open System Interconnection
OSP	Organizational Security Policies
PAE	Physical Address Extension
PC	Personal Computer
PKCS	Public Key Cryptography Standard
PoE	Power over Ethernet
POST	Power-on Self-Test
PP	Protection Profile
PRNG	Pseudo Random Number Generator
PSK	Pre-Shared Key
PUB	Publication
RA	Registration Authority
RADIUS	Remote Authentication Dial-In User Service
RCT	Repetition Count Test
RFC	Request for Comment
RJ	Registered Jack
RNG	Random Number Generator
ROM	Read-Only Memory
RSA	Rivest, Shamir and Adleman
SA	Security Association
SAK	Secure Association Key
SAR	Security Assurance Requirement
SATA	Serial Advanced Technology Attachment
SC	Secure Channel
SCI	Secure Channel Identifier
SCEP	Simple Certificate Enrollment Protocol
SCI	Secure Channel Identifier

Acronyms / Abbreviations	Definition
SecTAG	MAC Security TAG
SecY	MAC Security Entity
SFP	Small–Form-Factor Pluggable Port
SFR	Security Functional Requirement
SHA	Secure Hash Algorithm
SHS	Secure Hash Standard
SM	Service Module
SNMP	Simple Network Management Protocol
SP	Special Publication
SPD	Security Policy Definition
SSD	Solid State Drive
SSHv2	Secure Shell (version 2)
ST	Security Target
TAC	Technical Assistance Center
TCP	Transport Control Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
TLS	Transport Layer Security
TOE	Target of Evaluation
TSC	TSF Scope of Control
TSF	TOE Security Function
TSP	TOE Security Policy
UADP	Unified Access Data Plane
UDP	User Datagram Protocol
U.S.	United States
USB	Universal Serial Bus
UTP	Universal Twisted Pair
VAC	Volts of Alternating Current
VPN	Virtual Private Network
WAN	Wide Area Network
WIC	WAN Interface Card

10 Annex D: Terminology

Table 22 below provides a list of terms that are common and may be used in this Security Target.

Table 22 Terminology

Term	Definition
Authorized Administrator	Any user that has been assigned to a privilege level that is permitted to perform all TSF-related functions.
IOS-XE	Proprietary operating system developed by Cisco Systems.
MACsec Peer	This includes any MACsec peer with which the TOE participates in MACsec communications. MACsec Peer may be any device that supports MACsec communications
Packet	A block of data sent over the network transmitting the identities of the sending and receiving stations, error-control information, and message.
Security Administrator	Synonymous with Authorized Administrator for the purposes of this evaluation.
User	Any entity (human user or external IT entity) outside the TOE that interacts with the TOE.
vty	vty is a term used by Cisco to describe a single terminal (whereas Terminal is more of a verb or general action term).
Firmware (per NIST for FIPS validated cryptographic modules)	The programs and data components of a cryptographic module that are stored in hardware (e.g., ROM, PROM, EPROM, EEPROM or FLASH) within the cryptographic boundary and cannot be dynamically written or modified during execution.

11 Annex E: References

Documentation listed in Table 23 below was used to prepare this ST.

Table 23 References

Identifier	Description
[CC_PART1]	Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and general model, Version 3.1, Revision 5, dated: April 2017
[CC_PART2]	Common Criteria for Information Technology Security Evaluation – Part 2: Security functional components, Version 3.1, Revision 5, dated: April 2017
[CC_PART3]	Common Criteria for Information Technology Security Evaluation – Part 3: Security assurance components, Version 3.1, Revision 5, dated: April 2017
[CEM]	Common Methodology for Information Technology Security Evaluation – Evaluation Methodology, Version 3.1, Revision 5, dated: April 2017
[NDcPP]	collaborative Protection Profile for Network Devices, Version NDcPP v2.2e, 23 March 2020
[MACsec EP]	Network Device Collaborative Protection Profile (NDcPP) Extended Package MACsec Ethernet Encryption (MACsec EP), Version 1.2, 10 May 2016
[800-38B]	NIST Special Publication 800-38B, May 2005
[800-56Arev3]	NIST Special Publication 800-56Arev3, April 2018
[800-56Brev2]	NIST Special Publication 800-56Brev2 Recommendation for Pair-Wise, March 2019
[FIPS 140-2]	FIPS PUB 140-2 Federal Information Processing Standards Publication
[FIPS PUB 186-4]	FIPS PUB 186-4 Federal Information Processing Standards Publication Digital Signature Standard (DSS) October 2015
[800-90Arev1]	NIST Special Publication 800-90Arev1 Recommendation for Random Number Generation Using Deterministic Random Bit Generators June 2015
[800-90Brev1]	NIST Special Publication 800-90B Recommendation for the Entropy Sources Used for Random Bit Generation January 2018
[FIPS PUB 180-3]	FIPS PUB 180-3 Federal Information Processing Standards Publication Secure Hash Standard (SHS) October 2008