Cisco 8000 Series Routers running IOS-XR Version 7.3

Security Target

Version: 1.0
Date: October 21, 2022

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

Table of Contents

List of Tables

List of Figures

Acronyms

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

Table 1 Acronyms

Acronyms/Abbreviations	Definition
AES	Advanced Encryption Standard
BRI	Basic Rate Interface
CAVP	Cryptographic Algorithm Validation Program
CC	Common Criteria for Information Technology Security Evaluation
CEM	Common Evaluation Methodology for Information Technology Security
CM	Configuration Management
CSU	Channel Service Unit
DHCP	Dynamic Host Configuration Protocol
DSU	Data Service Unit
EHWIC	Ethernet High-Speed WIC
ESP	Encapsulating Security Payload
ESPr	Embedded Services Processors
GE	Gigabit Ethernet port
HTTPS	Hyper-Text Transport Protocol Secure
IT	Information Technology
NDcPP	collaborative Protection Profile for Network Devices
OS	Operating System
PoE	Power over Ethernet
PP	Protection Profile
SA	Security Association
SFP	Small–form-factor pluggable port
SHS	Secure Hash Standard
ST	Security Target
TCP	Transmission Control Protocol
TSC	TSF Scope of Control
TSF	TOE Security Function
TSP	TOE Security Policy
WAN	Wide Area Network
WIC	WAN Interface Card

Document Introduction

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

This document provides the basis for an evaluation of a specific Target of Evaluation (TOE), Cisco 8000 Series Routers (C8000). This Security Target (ST) defines a set of assumptions about the aspects of the environment, a list of threats that the product intends to counter, a set of security objectives, a set of security requirements, and the IT security functions provided by the TOE which meet the set of requirements.

1 Security Target Introduction

The Security Target contains the following sections:

- Security Target Introduction [Section 1]
- Conformance Claims [Section 2]
- Security Problem Definition [Section 3]
- Security Objectives [Section 4]
- IT Security Requirements [Section 5]
- TOE Summary Specification [Section 6]
- Annex A: Key Zeroization [Section 7]
- Annex B: NIAP Technical Decisions (TD) [Section 8]
- Annex C: References [Section 9]

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

1.1 ST and TOE Reference

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

Table 2 ST and TOE Identification

Name	Description
ST Title	Cisco 8000 Series Routers running IOS-XR Version 7.3
ST Version	1.0
Publication Date	October 21, 2022
Vendor and ST Author	Cisco Systems, Inc.
TOE Reference	Cisco 8000 Series Routers (C8000)
TOE Hardware Models	8808-SYS, 8812-SYS, 8818-SYS
TOE Software Version	IOS-XR 7.3
Keywords	Router, Network Appliance, Data Protection, Authentication, Cryptography, Secure Administration, Network Device, MACsec

1.2 TOE Overview

The Cisco 8000 Series Routers (herein after referred to as the C8000) is a purpose-built, routing platform that also supports MACsec encryption. The TOE includes the hardware models as defined in Table 4.

1.3 TOE Product Type

The TOE is a Network Device as defined in NDcPP 2.2e.

The C8000 delivers provider-class routing functionality at high density, performance, and power. This enables the C8000 to be deployed into a range of routing roles thus streamlining qualification, deployment, and operations. The C8000 comprises a full range of feature-rich, highly scalable, deep-buffered, 400G-optimized routers ranging from 10.8 Tbps to a rack-mountable modular system capable of 260 Tbps of full duplex, line rate forwarding.

The C8000 runs IOS-XR that is a microkernel based network operating system. IOS-XR is able to process data as it comes into the router without buffering delays. The microkernel is responsible for specific functions such as memory management, interrupt handling, scheduling, task switching, synchronization, and inter-process communication. The microkernel's functions do not include other system services such as device drivers, file system, and network stacks; those services are implemented as independent processes outside the kernel, and they can be restarted like any other application.

1.4 Supported non-TOE Hardware/ Software/ Firmware

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

Table 3 IT Environment Components

Component	Required	Usage/Purpose Description for TOE performance
Management Workstation with SSH Client	Yes	This includes any IT Environment Management workstation with a SSH client installed that is used by the TOE administrator to support TOE administration through SSH protected channels. Any SSH client that supports SSHv2 may be used.
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.
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.
Audit (syslog) Server	Yes	This includes any syslog server to which the TOE would transmit syslog messages. Also referred to as audit server in the ST
Certificate Authority	Yes	This includes any Operational Environment Certificate Authority on the TOE network. This can be used to provide the TOE with a valid certificate during certificate enrollment.

1.5 TOE Description

This section provides an overview of the C8000 Target of Evaluation (TOE). This section also defines the TOE components included in the evaluated configuration of the TOE. The TOE is comprised of both software and hardware. The hardware is comprised of the following: 8808-SYS, 8812-SYS and 8818-SYS. The software is comprised of the Cisco IOS-XR 7.3.

The TOE consists of a number of components including:

- Chassis: The TOE chassis includes 16 RU (8 slot), 21 RU (12 slot) and 33 RU (18 slot) form factors. The chassis is the component of the TOE in which all other TOE components are housed.
- Route Processor (RP): A route processor in each chassis provide the advanced routing capabilities of the TOE. They also monitor and manage the other components in the C8000.
- Fabric Cards: 8808-FC, 8812-FC and 8818-FC
- Supporting Line Cards: 8800-LC-48H and 8800-LC-36FH

1.6 TOE Evaluated Configuration

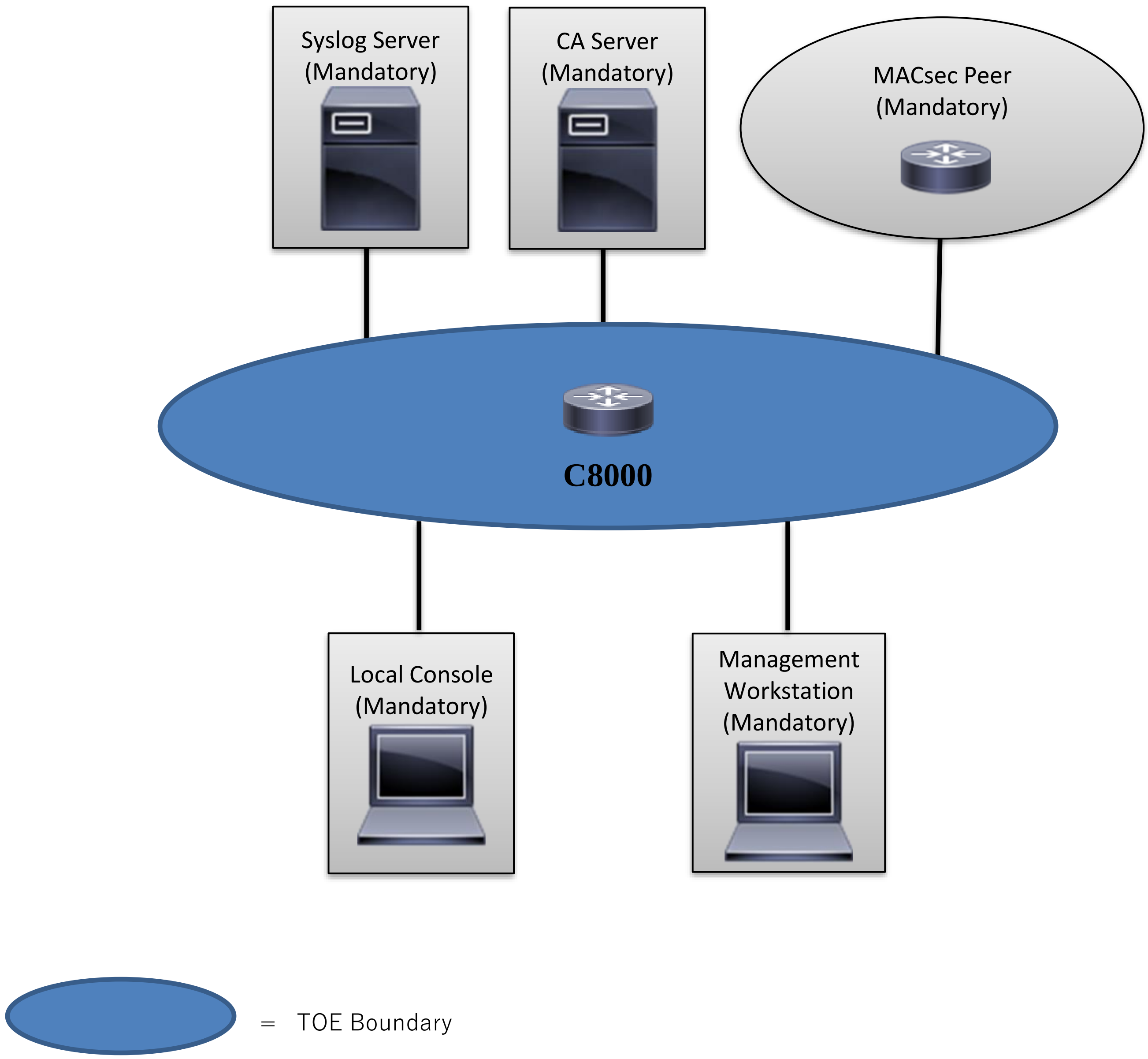
The TOE consists of one or more physical devices as specified in section 1.7 below along with MACsec-supporting hardware (8800-LC-48H) and includes the Cisco IOS-XR software. The TOE has two or more network interfaces and is connected to at least one internal and one external network. The Cisco IOS-XR configuration determines how packets are handled to and from the TOE's network interfaces. The router configuration will determine how traffic flows received on an interface will be handled. Typically, packet flows are passed through the internetworking device and forwarded to their configured destination.

An external syslog server must be used to store audit records. The TOE authenticates those devices with X.509v3 certificates and protects communication channels with the TLS protocol. Secure remote administration is protected with SSH which is implemented with authentication failure handling.

For remote administration, a secure session using SSHv2 must be established.

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

Figure 1 TOE Example Deployment



The previous figure includes the following:

- Examples of TOE Models
- The following are considered to be in the IT Environment:
    - MACsec Peer
    - Management Workstation
    - Audit (Syslog) Server
    - Local Console
    - Certificate Authority

NOTE: While the previous figure includes several non-TOE IT environment devices, the TOE is only the C8000 device. Only one TOE device is required for deployment in an evaluated configuration.

1.7 Physical Scope of the TOE

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

- Chassis: 8808-SYS, 8812-SYS and 8818-SYS
- Route Processors (RP): 8800-RP
- Fabric Cards: 8808-FC, 8812-FC and 8818-FC
- Supporting Line Cards: 8800-LC-48H and 8800-LC-36FH

The network, on which they reside, is considered part of the environment. The software is pre-installed and is comprised of the Cisco IOS-XR software image Release 7.3. In addition, the software image is also downloadable from the Cisco web site. A login id and password is required to download the software image. The TOE is comprised of the following physical specifications as described in Table 4 below:

Table 4 Hardware Models and Specifications

Hardware	Picture	Features
Cisco 8000 Series Routers (C8000) 8808-SYS 8812-SYS 8818-SYS 8800-RP 8808-FC 8812-FC 8818-FC 8800-LC-48H 8800-LC-36FH		**Physical dimensions** (H x W x D) • 8808: 28 x 17.45 x 33.73 in. (71.12 x 44.32 x 85.7 cm) – 16 RU – 8 line cards • 8812: 36.75 x 17.45 x 35.43 in. (93.345 x 44.23 x 90 cm) – 21 RU – 12 line cards • 8818: 57.75 x 17.45 x 35.43 in. (146.7 x 44.23 x 90 cm) – 33 RU – 18 line cards **Route Processors (RP)** • Intel Xeon D-1530 (Broadwell) CPU • 32 GB of DRAM • RS-232 console • 10 GbE SFP+ • 1 GbE Management • 2x USB2.0 **Interfaces** • 48 QSFP28 100 GbE • 36 QSFP56-DD 400 GbE **Power**

Hardware	Picture	Features
	 8800-LC-48H  8800-LC-36FH	• 8808 and 8812 – 9 high-voltage power supplies or 12 48V DC power supplies • 8818 - 18 high-voltage power supplies or 24 48V DC power supplies

1.8 Logical Scope of the TOE

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

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

1.8.1 Security Audit

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

1.8.2 Cryptographic Support

The TOE provides cryptography in support of other TOE security functionality. All the algorithms claimed have CAVP certificates (Operational Environment – Intel Xeon D-1530 (Broadwell)). In addition, the TOE supports MACsec using the CoMIRA Mentor Questa Sim 10.7 processor (see Table 5 for certificate references).

Table 5 FIPS References

Algorithm	Description	Supported Mode	Module	CAVP Cert. #	SFR
AES	Used for symmetric encryption/decryption	CBC (128 and 256) GCM (128 and 256) CTR (128 and 256) AES Key Wrap and CMAC (128 and 256)	FOM 6.2	A388	FCS_COP.1/DataEncryption FCS_COP.1(1)/KeyedHash:CMAC FCS_COP.1(2) Cryptographic Operation
		GCM (128 and 256)	MACsec	C1668	
SHS (SHA-1, SHA-256, SHA-512)	Cryptographic hashing services	Byte Oriented	FOM 6.2	A388	FCS_COP.1/Hash
HMAC (HMAC-SHA-1, HMAC-SHA-256, SHA-512)	Keyed hashing services and software integrity test	Byte Oriented	FOM 6.2	A388	FCS_COP.1/KeyedHash
DRBG	Deterministic random bit generation services in accordance with ISO/IEC 18031:2011	CTR_DRBG (AES 256)	FOM 6.2	A388	FCS_RBG_EXT.1
RSA	Signature Verification and key transport	PKCS#1 v.1.5, 3072 bit key, FIPS 186-4 Key Gen	FOM 6.2	A388	FCS_CKM.1 FCS_COP.1/SigGen

The TOE provides cryptography in support of remote administrative management via SSHv2 and secures the session between the C8000 and remote syslog server using TLS.

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 6 below:

Table 6 TOE Provided Cryptography

Cryptographic Method	Use within the TOE
Secure Shell Establishment	Used to establish initial SSH session.
RSA Signature Services	Used in SSH session establishment. Used in TLS session establishment. X.509 certificate signing.
SHS	Used to provide SSH traffic integrity verification Used for keyed-hash message authentication
AES	Used to encrypt SSH session traffic. Used to encrypt TLS session traffic. Used to encrypt MACsec traffic.
RSA	Used to provide cryptographic signature services
HMAC	Used for keyed hash, integrity services in SSH session establishment.
TLS	Used to secure traffic to the syslog server.

1.8.3 Identification and authentication

The TOE provides authentication services for administrative users wishing to connect to the TOEs secure CLI administrator interface. The TOE requires Authorized Administrators to authenticate prior to being granted access to any of the management functionality. The TOE can be configured to require a minimum password length of 15 characters as well as mandatory password complexity rules.

After a configurable number of incorrect login attempts, C8000 will lockout the account until a configured amount of time for lockout expires.

The TOE provides administrator authentication against a local user database. Password-based authentication can be performed on the serial console or SSH interfaces. The SSHv2 interface also supports authentication using SSH keys.

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

1.8.4 Security Management

The TOE provides secure administrative services for management of general TOE configuration and the security functionality provided by the TOE. All TOE administration occurs either through a secure SSHv2 session or via a local console connection. The TOE provides the ability to securely manage all TOE administrative users, all identification and authentication, all audit functionality of the TOE, all TOE cryptographic functionality, the timestamps maintained by the TOE, and updates to the TOE. The TOE supports a privileged administrator role. Only the privileged administrator can perform the above security relevant management functions.

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

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

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

The TOE internally maintains the date and time. This date and time is used as the timestamp that is applied to audit records generated by the TOE. Administrators can update the TOE's clock manually. Finally, the TOE performs testing to verify correct operation of the router itself and that of the cryptographic module.

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

1.8.6 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 a Security Administrator specified banner on the CLI management interface prior to allowing any administrative access to the TOE.

1.8.7 Trusted path/Channels

The TOE establishes a trusted path between the appliance and the CLI using SSHv2 and the syslog server using TLS. MACsec is used to secure communication channels between MACsec peers at Layer 2.

1.9 Excluded Functionality

The following functionality is excluded from the evaluation:

Table 7 Excluded Functionality

Excluded Functionality	Exclusion Rationale
Non-FIPS 140-2 mode of operation	This mode of operation includes non-FIPS allowed operations.

These services will be disabled by configuration settings. The exclusion of this functionality does not affect compliance to the NDcPP v2.2e and 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 8 below. This ST applies the NIAP Technical Decisions as described in Table 20.

Table 8 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 (MACSECEP)	1.2	May 10, 2016

2.3 Protection Profile Conformance Claim Rationale

2.3.1 TOE Appropriateness

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

- collaborative Protection Profile for Network Devices (NDcPP) Version 2.2e
- Network Device collaborative Protection Profile (NDcPP) Extended Package MACsec Ethernet Encryption (MACSECEP), Version 1.2

2.3.2 TOE Security Problem Definition Consistency

The Assumptions, Threats, and Organizational Security Policies included in the Security Target represent the Assumptions, Threats, and Organizational Security Policies specified in the collaborative Protection Profile for Network Devices (NDcPP) Version 2.2e and MACsec Ethernet Encryption (MACSECEP) Version 1.2 for which conformance is claimed verbatim. All concepts covered in the Protection Profile Security Problem Definition are included in the Security Target Statement of Security Objectives Consistency.

The Security Objectives included in the Security Target represent the Security Objectives specified in the NDcPP Version 2.2e and MACSECEP v1.2 for which conformance is claimed verbatim. All concepts covered in the Protection Profile's Statement of Security Objectives are included in the Security Target.

2.3.3 Statement of Security Requirements Consistency

The Security Functional Requirements included in the Security Target represent the Security Functional Requirements specified in the NDcPP v2.2e and MACSECEP v1.2 for which conformance is claimed verbatim. All concepts covered in the Protection Profile's Statement of Security Requirements are included in this Security Target. Additionally, the Security Assurance Requirements included in this Security Target are identical to the Security Assurance Requirements included in the NDcPP Version 2.2e and the MACSECEP v1.2.

3 Security Problem Definition

This chapter identifies the following:

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

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

3.1 Assumptions

The specific conditions listed in the following subsections are assumed to exist in the TOE's environment. These assumptions include both practical realities in the development of the TOE security requirements and the essential environmental conditions on the use of the TOE.

Table 9 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 or interfere with the device's physical interconnections and correct operation. This protection is assumed to be sufficient to protect the device and the data it contains. As a result, the cPP does not include any requirements on physical tamper protection or other physical attack mitigations. The cPP does not expect the product to defend against physical access to the device that allows unauthorized entities to extract data, bypass other controls, or otherwise manipulate the device. For vNDs, this assumption applies to the physical platform on which the VM runs.
A.LIMITED_FUNCTIONALITY	The device is assumed to provide networking functionality as its core function and not provide functionality/ services that could be deemed as general purpose computing. For example the device should not provide computing platform for general purpose applications (unrelated to networking functionality). In the case of vNDs, the VS is considered part of the TOE with only one vND instance for each physical hardware platform. The exception being where components of the distributed TOE run inside more than one virtual machine (VM) on a single VS. There are no other guest VMs on the physical platform providing non-Network Device 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~~

Assumption	Assumption Definition
	~~covered by the ND cPP. It is assumed that this protection will be covered by cPPs for particular types of network devices (e.g, firewall).~~ [1]
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 being appropriately trained, following policy, and adhering to guidance documentation. Administrators are trusted to ensure passwords/credentials have sufficient strength and entropy and to lack malicious intent when administering the device. The Network Device is not expected to be capable of defending against a malicious administrator that actively works to bypass or compromise the security of the device. For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are expected to fully validate (e.g. offline verification) any CA certificate (root CA certificate or intermediate CA certificate) loaded into the TOE's trust store (aka 'root store', ' trusted CA Key Store', or similar) as a trust anchor prior to use (e.g. offline verification).
A.REGULAR_UPDATES	The Network Device firmware and software is assumed to be updated by an administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
A.ADMIN_CREDENTIALS_SECURE	The administrator's credentials (private key) used to access the Network Device are protected by the platform on which they reside.
A.RESIDUAL_INFORMATION	The Administrator must ensure that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment.

3.2 Threats

The following table lists the threats addressed by the TOE and the IT Environment. The assumed level of expertise of the attacker for all the threats identified below is Enhanced-Basic.

Table 10 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

[1] Assumption does not apply per MACsec EP v1.2

	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.WEAK_AUTHENTICATION_ENDPOINTS	Threat agents may take advantage of secure protocols that use weak methods to authenticate the endpoints – e.g., shared password that is guessable or transported as plaintext. The consequences are the same as a poorly designed protocol, the attacker could masquerade as the administrator or another device, and the attacker could insert themselves into the network stream and perform a man-in-the-middle attack. The result is the critical network traffic is exposed and there could be a loss of confidentiality and integrity, and potentially the Network Device itself could be compromised.
T.UPDATE_COMPROMISE	Threat agents may attempt to provide a compromised update of the software or firmware which undermines the security functionality of the device. 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 include replacing existing credentials with an attacker's credentials, modifying existing credentials, or obtaining the administrator or device credentials for use by the attacker.
T.PASSWORD_CRACKING	Threat agents may be able to take advantage of weak administrative passwords to gain privileged access to the device. Having privileged access to the device provides the attacker unfettered access to the network traffic, and may allow them to take advantage of any trust relationships with other Network Devices.
T.SECURITY_FUNCTIONALITY_FAILURE	An external, unauthorized entity could make use of failed or compromised security functionality and might therefore subsequently use or abuse security functions without prior authentication to access, change or modify device data, critical network traffic or security functionality of the device.

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 11 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.

This document identifies objectives of the TOE as O.objective with objective specifying a unique name. Objectives that apply to the IT environment are designated as OE.objective with objective specifying a unique name.

4.1 Security Objectives for the TOE

The following table, Security Objectives for the TOE, identifies the security objectives of the TOE. These security objectives reflect the stated intent to counter identified threats and/or comply with any security policies. The security objectives below have been drawn verbatim from [NDcPPv2.2e].

Table 12 Security Objectives for the TOE

TOE Objective	TOE Security Objective Definition
O.CRYPTOGRAPHIC_FUNCTIONS	The TOE will provide cryptographic functions that are used to establish secure communications channels between the TOE and the Operational Environment.
O.AUTHENTICATION	The TOE will provide the ability to establish connectivity associations with other MACsec peers.
O.PORT_FILTERING	The TOE will provide the ability to restrict the flow of traffic between networks based on originating port and established connection information.
O.SYSTEM_MONITORING	The TOE will provide the means to detect when security-relevant events occur and generate audit events in response to this detection.
O.AUTHORIZED_ADMINISTRATION	The TOE will provide management functions that can be used to securely manage the TSF.
O.TSF_INTEGRITY	The TOE will provide mechanisms to ensure that it only operates when its integrity is verified.
O.REPLAY_DETECTION	The TOE will provide the means to detect attempted replay of MACsec traffic by inspection of packet header information.
O.VERIFIABLE_UPDATES	The TOE will provide a mechanism to verify the authenticity and integrity of product updates before they are applied.

4.2 Security Objectives for the Environment

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

Table 13 Security Objectives for the Environment

Environment Security Objective	IT Environment Security Objective Definition
OE.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.
OE.NO_GENERAL_PURPOSE	There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE. Note: For vNDs the TOE includes only the contents of the its own VM, and does not include other VMs or the VS.
~~OE.NO_THRU_TRAFFIC_PROTECTION~~	~~The TOE does not provide any protection of traffic that traverses it. It is assumed that protection of this traffic will be covered by other security and assurance measures in the operational environment.~~ [2]
OE.TRUSTED_ADMIN	Security Administrators are trusted to follow and apply all guidance documentation in a trusted manner. For vNDs, this includes the VS Administrator responsible for configuring the VMs that implement ND functionality. For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are assumed to monitor the revocation status of all certificates in the TOE's trust store and to remove any certificate from the TOE's trust store in case such certificate can no longer be trusted.
OE.UPDATES	The TOE firmware and software is updated by an administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
OE.ADMIN_CREDENTIALS_SECURE	The administrator's credentials (private key) used to access the TOE must be protected on any other platform on which they reside.
OE.RESIDUAL_INFORMATION	The Security Administrator ensures that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment. For vNDs, this applies when the physical platform on which the VM runs is removed from its operational environment.

[2] Assumption does not apply per MACsec EP v1.2

5 Security Requirements

This section identifies the Security Functional Requirements for the TOE. The Security Functional Requirements included in this section are derived from Part 2 of the *Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5, dated: April 2017* and all international interpretations.

5.1 Conventions

The CC defines operations on Security Functional Requirements: assignments, selections, assignments within selections and refinements. This document uses the following font conventions to identify the operations defined by the CC:

- Assignment: Indicated with *italicized* text;
- Assignment completed within a selection in the cPP: the completed assignment text is indicated with <u>*italicized and underlined text*</u>
- Refinement: Indicated with **bold** text;
- Selection: Indicated with <u>underlined</u> text;
- Iteration: Indicated by appending the iteration number in parenthesis, e.g., (1), (2), (3).
- Where operations were completed in the NDcPP itself, the formatting used in the NDcPP has been retained.

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

- All SFRs from MACSECEP reproduced as-is
- SFRs that appear in both NDcPP and MACSECEP are modified based on instructions specified in MACSECEP unless the NDcPP has a stricter and more prescriptive version of the SFR.

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 14 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
	FCS_CKM.2	Cryptographic Key Establishment
	FCS_CKM.4	Cryptographic Key Destruction
	FCS_COP.1/DataEncryption	Cryptographic Operation (for data encryption/decryption)
	FCS_COP.1/SigGen	Cryptographic Operation (for cryptographic signature)
	FCS_COP.1/Hash	Cryptographic Operation (for cryptographic hashing)
	FCS_COP.1KeyedHash	Cryptographic Operation (for keyed-hash message authentication)

Class Name	Component Identification	Component Name
	FCS_COP.1(1)	KeyedHashCMAC Cryptographic Operation (AES-CMAC Keyed Hash Algorithm)
	FCS_COP.1(2)	Cryptographic Operation (MACsec Data Encryption/Decryption)
	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_TLSC_EXT.1	TLS Client Protocol
	FCS_RBG_EXT.1	Random Bit Generation
FIA: Identification and authentication	FIA_AFL.1	Authentication Failure Management
	FIA_PMG_EXT.1	Password Management
	FIA_PSK_EXT.1	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 Validiation
	FIA_X509_EXT.2	X.509 Certificate Authentication
	FIA_X509_EXT.3	X.509 Certificate Requests
FMT: Security management	FMT_MOF.1/Services	Trusted Update - Management of TSF Data
	FMT_MOF.1/ManualUpdate	Trusted Update - Management of security functions behaviour
	FMT_MTD.1/CryptoKeys	Management of TSF Data
	FMT_MTD.1/CoreData	Management of TSF Data
	FMT_SMF.1	Specification of Management Functions
	FMT_SMR.2	Restrictions on security roles

Class Name	Component Identification	Component Name
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	Fail Secure
	FPT_RPL.1	Replay Detection
	FPT_SKP_EXT.1	Protection of TSF Data (for reading of all symmetric keys)
	FPT_STM_EXT.1	Reliable Time Stamps
	FPT_TST_EXT.1	Extended: TSF Testing
	FPT_TUD_EXT.1	Extended: 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

5.3 SFRs from NDcPP and MACsec EP

5.3.1 Security audit (FAU)

5.3.1.1 FAU_GEN.1 Audit data generation

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

a) Start-up and shut-down of the audit functions;

b) All auditable events for the <u>not specified</u> level of audit; and

c) All administrator actions comprising:

- *Administrative login and logout (name of user account shall be logged if individual user accounts are required for administrators).*
- *Changes to TSF data related to configuration changes (in addition to the information that a change occurred it shall be logged what has been changed).*
- *Generating/import of, changing, or deleting of cryptographic keys (in addition to the action itself a unique key name or key reference shall be logged).*
- *Resetting passwords (name of related user account shall be logged).*
- *[<u>no other actions</u>];*

d) Specifically defined auditable events listed in Table 15.

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

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

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 15.*

Table 15 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 Cryptographic Operation (AES-CMAC Keyed Hash Algorithm)	None	None
FCS_COP.1(2) Cryptographic Operation (MACsec Data Encryption/Decryption)	None.	None.
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_RBG_EXT.1	None.	None.
FCS_SSHS_EXT.1	Failure to establish an SSH session	Reason for failure.
FCS_TLSC_EXT.1	Failure to establish an TLS session	Reason for failure.
FIA_AFL.1	Unsuccessful login attempts limit is met or exceeded. Administrator lockout due to excessive authentication failures	Origin of the attempt (e.g., IP address)

SFR	Auditable Event	Additional Audit Record Contents
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_MOF.1/Services	Starting and stopping of Services	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_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_TUD_EXT.1	Initiation of update. result of the update attempt (success or failure)	None.
FPT_TST_EXT.1	None.	None.

SFR	Auditable Event	Additional Audit Record Contents
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. Failure of the trusted path functions.	None.

5.3.1.2 FAU_GEN.2 User Identity Association

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

5.3.1.3 FAU_STG_EXT.1 Protected Audit Event Storage

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

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

- 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.

5.3.2 Cryptographic Support (FCS)

5.3.2.1 FCS_CKM.1 Cryptographic Key Generation

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

- RSA schemes using cryptographic key sizes of 2048-bit 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, RFC-7919]

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

5.3.2.2 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, groups listed in RFC-7919];

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

5.3.2.3 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]*];
- *For plaintext keys in non-volatile storage, the destruction shall be executed by the invocation of an interface provided by a part of the TSF that* [
  - logically addresses the storage location of the key and performs a [single] overwrite consisting of *[zeroes]];*

that meets the following: *No Standard*.

5.3.2.4 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, CTR, GCM] *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, CTR as specified in ISO 10116, GCM as specified in ISO 19772].

5.3.2.5 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) [*3072 bits*],

~~*]* and cryptographic key sizes [assignment: *cryptographic key sizes*]~~

that meet the following: *[*

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

].

5.3.2.6 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*.

5.3.2.7 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".*

5.3.2.8 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, 256 bits**] and message digest size **of 128** bits that meets **NIST SP 800-38B.**

5.3.2.9 FCS_COP.1 (2) Cryptographic Operation (MACsec Data Encryption/Decryption)

FCS_COP.1.1(2) 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, 256 bits]** that meet the following: **AES as specified in ISO 18033-3, AES Key Wrap as specified in NIST SP 800-38F, GCM as specified in ISO 19772.**

5.3.2.10 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.

5.3.2.11 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.

5.3.2.12 FCS_ MACSEC_EXT.3 MACsec Randomness

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

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

5.3.2.13 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 (CKN) 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).

5.3.2.14 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 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.

5.3.2.15 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.

5.3.2.16 FCS_SSHS_EXT.1 SSH Server Protocol

FCS_SSHS_EXT.1.1 The TSF shall implement the SSH protocol in accordance with RFCs 4251, 4252, 4253, 4254 [4344, 6668].

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

FCS_SSHS_EXT.1.3 The TSF shall ensure that, as described in RFC 4253, packets greater than [*65,535 bytes*] bytes in an SSH transport connection are dropped.

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

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

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

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

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

5.3.2.17 FCS_TLSC_EXT.1 – TLS Client Protocol

FCS_TLSC_EXT.1.1 The TSF shall implement [TLS 1.2 (RFC 5246)] and reject all other TLS and SSL versions. The TLS implementation will support the following ciphersuites:

[

- TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268
- TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268] *and no other ciphersuites.*

FCS_TLSC_EXT.1.2 The TSF shall verify that the presented identifier matches: [the reference identifiers defined in RFC 6125 section 6, and no other attribute types].

FCS_TLSC_EXT.1.3 When establishing a trusted channel, by default the TSF shall not establish a trusted channel if the server certificate is invalid. The TSF shall also [Not implement any administrator override mechanism].

FCS_TLSC_EXT.1.4 The TSF shall [not *present the Supported Elliptic Curves/Supported Groups Extension*] **in the Client Hello**.

5.3.3 Identification and authentication (FIA)

5.3.3.1 FIA_AFL.1 Authentication Failure Management

FIA_AFL.1.1 The TSF shall detect when an Administrator configurable positive integer within [*1 to 24*] unsuccessful authentication attempts occur related to *Administrators attempting to authenticate remotely using a password*.

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been met, the TSF shall [prevent the offending Administrator from successfully establishing remote session using any authentication method that involves a password until *an Administrator defined time period has elapsed].*

5.3.3.2 FIA_PMG_EXT.1 Password Management

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

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

b) Minimum password length shall be configurable to between [*15*] and [*15*] characters.

5.3.3.3 FIA_ PSK_EXT.1(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, [*no other protocols*].

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

5.3.3.4 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.

5.3.3.5 FIA_UAU_EXT.2 Password-based Authentication Mechanism

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

5.3.3.6 FIA_UAU.7 Protected Authentication Feedback

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

5.3.3.7 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 certificate path validation **supporting a minimum path length of three certificates.**
- The certificate path must terminate with a trusted CA certificate designated as a trust anchor.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of the certificate using [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.

5.3.3.8 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 [TLS], and [no additional uses].

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

5.3.3.9 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].

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

5.3.4 Security management (FMT)

5.3.4.1 FMT_MOF.1/ManualUpdate Management of Security Functions Behavior

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

5.3.4.2 FMT_MOF. 1/ Services Management of Security Functions Behavior

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

5.3.4.3 FMT_MTD.1/CoreData Management of TSF Data

FMT_MTD.1/CoreData The TSF shall restrict the ability to *manage* the *TSF data* to *Security Administrators*.

5.3.4.4 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*.

5.3.4.5 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, hash comparison] capability prior to installing those updates;*
- *Ability to configure the authentication failure parameters for FIA_AFL.1;*
- Ability of a Security Administrator to:
    - *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*

[
- *Ability to start and stop services]*
- *Ability to configure audit behavior (e.g. changes to storage locations for audit; changes to behaviour when local audit storage space is full);*
- Ability to configure the list of TOE-provided services available before an entity is identified and authenticated, as specified in FIA_UIA_EXT.1;
- Ability to configure the cryptographic functionality;
- Ability to configure thresholds for SSH rekeying;
- *Ability to set the time which is used for time-stamps;*
- *Ability to configure the reference identifier for the peer;*
- *Ability to import X.509v3 certificates to the TOE'S trust store;*
- *Ability to manage the trusted public keys database;*
- *Configure the time interval for administrator lockout due to excessive authentication failures;*

]

5.3.4.6 FMT_SMR.2 Restrictions on Security Roles

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

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

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

are satisfied.

5.3.5 Protection of the TSF (FPT)

5.3.5.1 FPT_APW_EXT.1 Extended: Protection of Administrator Passwords

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

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

5.3.5.2 FPT_ CAK_EXT.1 Protection of CAK Data

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

5.3.5.3 FPT_ FLS.1(2)/SelfTest Failure with Preservation of Secure State

FPT_FLS.1.1(2)/SelfTest Refinement: The TSF shall **shut down** when the following types of failures occur: **failure of the power-on self-tests**, **failure of integrity check of the TSF executable image, failure of noise source health tests**.

5.3.5.4 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.

5.3.5.5 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.

5.3.5.6 FPT_STM_EXT.1 Reliable Time Stamps

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

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

5.3.5.7 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*
- *RSA Signature Known Answer Test (both signature/verification)*
- *RNG/DRBG Known Answer Test*
- *HMAC Known Answer Test*
- *SHA-1/256 Known Answer Test*
- *Software Integrity Test*

].

5.3.5.8 FPT_TUD_EXT.1 Extended: Trusted Update

FPT_TUD_EXT.1.1 The TSF shall provide *Security Administrators* the ability to query the currently executing version of the TOE firmware/software and [the most recently installed version of the TOE firmware/software].

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 a means to authenticate firmware/software updates to the TOE using a [digital signature, published hash**]** prior to installing those updates.

5.3.6 TOE Access (FTA)

5.3.6.1 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.

5.3.6.2 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*.

5.3.6.3 FTA_SSL.4 User-initiated Termination

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

5.3.6.4 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.

5.3.7 Trusted Path/Channels (FTP)

5.3.7.1 FTP_ITC.1 Inter-TSF trusted channel

FTP_ITC.1.1 Refinement: The TSF shall be capable of using [TLS, **MACsec] to** provide a trusted communication channel between itself and **authorized IT entities supporting the following capabilities: audit 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 *[communications with the following:*

- *external audit servers using TLS,*
- *MACsec peers using MACsec].*

5.3.7.2 FTP_TRP.1/Admin Trusted Path

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

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

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

5.4 TOE SFR Dependencies Rationale for SFRs Found in PP

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

5.5 Security Assurance Requirements

5.5.1 SAR Requirements

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

Table 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 User guidance
Life cycle support (ALC)	ALC_CMC.1	Labeling of the TOE
	ALC_CMS.1	TOE CM coverage
Tests (ATE)	ATE_IND.1	Independent testing - conformance
Vulnerability assessment (AVA)	AVA_VAN.1	Vulnerability analysis

5.5.2 Security Assurance Requirements Rationale

The Security Assurance Requirements (SARs) in this Security Target represent the SARs identified in the NDcPPv2.2e and MACSECEP v1.2. As such, the NDcPP SAR rationale is deemed acceptable since the PPs have been validated.

5.6 Assurance Measures

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

Table 17 Assurance Measures

Component	How requirement will be met
ADV_FSP.1	The functional specification describes the external interfaces of the TOE; such as the means for a user to invoke a service and the corresponding response of those services. The description includes the interface(s) that enforces a security functional requirement, the interface(s) that supports the enforcement of a security functional requirement, and the interface(s) that does not enforce any security functional requirements. The interfaces are described in terms of their purpose (general goal of the interface), method of use (how the interface is to be used), parameters (explicit inputs to and outputs from an interface that control the behavior of that interface), parameter descriptions (tells what the parameter is in some meaningful way), and error messages (identifies the condition that generated it, what the message is, and the meaning of any error codes). The development evidence also contains a tracing of the interfaces to the SFRs described in this ST.
AGD_OPE.1	The Administrative Guide provides the descriptions of the processes and procedures of how the administrative users of the TOE can securely administer the TOE using the interfaces that provide the features and functions detailed in the guidance.
AGD_PRE.1	The Installation Guide describes the installation, generation, and startup procedures so that the users of the TOE can put the components of the TOE in the evaluated configuration.
ALC_CMC.1 ALC_CMS.1	The Configuration Management (CM) document(s) describes how the consumer (end-user) of the TOE can identify the evaluated TOE (Target of Evaluation). The CM document(s), identifies the configuration items, how those configuration items are uniquely identified, and the adequacy of the procedures that are used to control and track changes that are made to the TOE. This includes details on what changes are tracked, how potential changes are incorporated, and the degree to which automation is used to reduce the scope for error. The TOE will also be provided along with the appropriate administrative guidance.
ATE_IND.1	Cisco will provide the TOE for testing.
AVA_VAN.1	Cisco will provide the TOE for testing.

6 TOE Summary Specification

6.1 TOE Security Functional Requirement Measures

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

Table 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: startup and shutdown of the audit mechanism cryptography related events, identification and authentication related events, and administrative events (the specific events and the contents of each audit record are listed in the table within the FAU_GEN.1 SFR, "Auditable Events Table"). Each of the events is specified in syslog records in enough detail to identify the user for which the event is associated, when the event occurred, where the event occurred, the outcome of the event, and the type of event that occurred such as generating keys, including the type of key and a key reference. Additionally, the startup and shutdown of the audit functionality is audited. The audit trail consists of the individual audit records; one audit record for each event that occurred. The audit record can contain up to 80 characters and a percent sign (%), which follows the time-stamp information. As noted above, the information includes at least all of the required information. Example audit events are included below: RP/0/RP0/CPU0:Nov 14 15:44:16.365 EST: exec[67642]: %SECURITY-LOGIN-6-AUTHEN_SUCCESS : Successfully authenticated user 'admin' from 'console' on 'con0_RP0_CPU0' In the above log events a date and timestamp is displayed as well as an event description. The log buffer is circular, so newer messages overwrite older messages after the buffer is full. Administrators are instructed to monitor the log buffer using the show logging command to view the audit records. The first message displayed is the oldest message in the buffer. There are other associated commands to clear the buffer, to set the logging level, etc. The administrator can set the level of the audit records to be displayed on the console or sent to the syslog server. For instance, all emergency, alerts, critical, errors, and warning messages can be sent to the console alerting the administrator that some action needs to be taken as these types of messages mean that the functionality of the TOE is affected. All notifications and information type message can be sent to the syslog server.
FAU_GEN.2	The TOE shall ensure that each auditable event is associated with the user that triggered the event and as a result, they are traceable to a specific user. For example, a human user, user identity or related session ID would be included in the audit record. For an IT entity or device, the IP address, MAC address, host name, or other configured identification is presented. A sample audit record is below: RP/0/RP0/CPU0:Dec 4 14:32:34.459 EST: config[67725]: %MGBL-CONFIG-6-DB_COMMIT : Configuration committed by user 'admin'. Use 'show configuration commit changes 1000000104' to view the changes.

TOE SFRs	How the SFR is Met
	RP/0/RP0/CPU0:SPITFIRE(config)#show configuration commit changes 1000000104 Building configuration... !! IOS XR Configuration logging buffered debugging RP/0/RP0/CPU0:Nov 14 15:49:25.467 EST: config[65679]: %MGBL-SYS-5-CONFIG_I : Configured from console by admin
FAU_STG_EXT.1	The TOE is a standalone TOE configured to export syslog records to a specified, external syslog server in real-time. The TOE protects communications with an external syslog server via TLS. If the connection fails, the TOE will store audit records on the TOE when it discovers it can no longer communicate with its configured syslog server. When the connection is restored, the TOE will transmit the buffer contents when connected to the syslog server. 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 2097152 to 125000000 bytes of available disk space. Only Authorized Administrators are able to clear the local logs, and local audit records are stored in a directory that does not allow administrators to modify the contents.
FCS_CKM.1 FCS_CKM.2	The TOE implements DH group 14 key establishment schemes that meets RFC 3526, Section 3 and RFC7919. The TOE acts as both a sender and receiver for Diffie-Helman based key establishment schemes. The TOE complies with section 5.6 and all subsections regarding asymmetric key pair generation and key establishment in the NIST SP 800-56A and with section 6. The TOE can create an RSA public-private key pair, with a minimum RSA key size of 2048-bit. RSA scheme can be used to generate a Certificate Signing Request (CSR). Via offline CSR or Simple Certificate Enrollment Protocol (SCEP), the TOE can: send the CSR to a Certificate Authority (CA) for the CA to generate a certificate; and receive its X.509 certificate from the CA. Integrity of the CSR and certificate during transit are assured through use of digital signatures (encrypting the hash of the TOE's public key contained in the CSR and certificate). The TOE can store and distribute the certificate to external entities including Registration Authorities (RA). The IOS-XR Software supports embedded PKI client functions that provide secure mechanisms for distributing, managing, and revoking certificates. The TOE can also use X.509v3 certificates for authentication of TLS sessions. The TOE acts as both a sender and receiver for RSA -based key establishment schemes. The RSA key establishment meets the 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. **Scheme** | **SFR** | **Service**

<table>
<tr><th>TOE SFRs</th><th>How the SFR is Met</th></tr>
<tr><td></td><td>
<table>
<tr><td>RSA
Key generation
Key establishment</td><td>FCS_SSHS_EXT.1</td><td>SSH Remote Administration</td></tr>
<tr><td></td><td>FCS_TLSC_EXT.1</td><td>Support for SSH and TLS key establishment</td></tr>
<tr><td>FFC
Key generation
Key establishment</td><td>FCS_SSHS_EXT.1</td><td>SSH Remote Administration</td></tr>
<tr><td></td><td>FIA_X509_EXT.1/Rev
FIA_X509_EXT.2
FIA_X509_EXT.3</td><td>Transmit generated audit data to an external IT entity</td></tr>
</table>
</td></tr>
<tr><td>FCS_CKM.4</td><td>The TOE meets all requirements specified in FIPS 140-2 for destruction of keys and Critical Security Parameters (CSPs) when no longer required for use.

See Table 19: TOE Key Zeroization in Section 7 Key Zeroization. The information provided in the table includes all of the all secrets, keys and associated values, the description, and the method used to zeroization when no longer required for use.

The information is provided in the reference section for ease and readability of all of the all secrets, keys and associated values, their description and zeroization methods.</td></tr>
<tr><td>FCS_COP.1/DataEncryption</td><td>The TOE provides symmetric encryption and decryption capabilities using AES in GCM, CTR and CBC mode (128 and 256 bits) as described in ISO 19772 and ISO 10116 respectively. Please see CAVP certificate in Table 5 for validation details. AES is implemented in the SSH protocol. The TOE provides AES encryption and decryption in support of SSHv2 for secure communications. Management of the cryptographic algorithms is provided through the CLI with auditing of those commands.

Please refer to Table 5 for all the CAVP references.</td></tr>
<tr><td>FCS_COP.1/SigGen</td><td>The TOE provides cryptographic signature services using the following: RSA Digital Signature Algorithm with key size of 3072 as specified in FIPS PUB 186-4, “Digital Signature Standard”.

The TOE provides cryptographic signatures in support of SSH and TLS for secure communications. Management of the cryptographic algorithms is provided through the CLI with auditing of those commands. The TOE provides the RSA option in support of SSH key establishment. RSA (3072-bits) is used in the establishment of SSHv2 and TLS key establishment. For SSH, RSA host keys are supported.

Please refer to Table 5 for all the CAVP references.</td></tr>
<tr><td>FCS_COP.1/Hash</td><td rowspan="2">The TOE provides cryptographic hashing services using SHA-1, SHA-256, and SHA-512 as specified in ISO/IEC 10118-3:2004.

The TOE provides Secure Hash Standard (SHS) hashing in support of SSH and TLS for secure communications. Management of the cryptographic algorithms is provided through the CLI with auditing of those commands.

The TOE provides keyed-hashing message authentication services using HMAC-SHA1, HMAC-SHA-256, and HMAC-SHA-512 with key sizes 160, 256, and 512 bits, and</td></tr>
<tr><td>FCS_COP.1/KeyedHash</td></tr>
</table>

TOE SFRs	How the SFR is Met
	message digest size 160, 256, and 512 bits as specified in ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2". The TOE provides SHS hashing and HMAC message authentication in support of SSHv2 and TLSv1.2 for secure communications. Management of the cryptographic algorithms is provided through the CLI with auditing of those commands. SHS hashing and HMAC message authentication (SHA-1) is used in the establishment of TLS and SSHv2 sessions. Please see CAVP certificate in Table 5 for validation details.
FCS_COP.1(1)/KeyedHashCMAC Cryptographic Operation (AES-CMAC Keyed Hash Algorithm) FCS_COP.1(2) Cryptographic Operation (MACsec Data Encryption/Decryption)	The TOE provides keyed-hash message authentication in accordance with AES-CMAC and cryptographic key sizes 128 and 256 bits with message digest size of 128 bits, block size of 128 bits, and MAC length of 128 bits which that meets NIST SP 800-38B. The TOE provides symmetric encryption and decryption capabilities using AES in AES Key Wrap and GCM mode (128 and 256 bits) as described in AES as specified in ISO 18033-3, AES Key Wrap in CMAC mode as specified in NIST SP 800-38F, GCM as specified in ISO 19772. AES is implemented in MACsec protocol. The relevant FIPS certificate numbers are listed in Table 5 FIPS References.
FCS_RBG_EXT.1	The TOE implements a NIST-approved AES-CTR Deterministic Random Bit Generator (DRBG), as specified in ISO/IEC 18031:2011 seeded by an entropy source that accumulates entropy from a TSF-hardware based noise source. The deterministic RBG is seeded with a minimum of 256 bits of entropy, which is at least equal to the greatest security strength of the keys and hashes that it will generate.
FCS_MACSEC_EXT.1	The TOE implements MACsec in compliance with IEEE Standard 802.1AE-2006. The MACsec connections maintain confidentiality of transmitted data and takes measures against frames transmitted or modified by unauthorized devices. In addition, the TOE implementation provides configuration options and management of the MACsec functionality, The SCI is composed of a globally unique 48-bit MAC Address and the Secure System Address (port). The SCI is part of the SecTAG if the SC bit is set and will be at the end of the tag. Any MPDUs during a given session that contain an SCI other than the one used to establish that session is rejected. Only EAPOL (PAE EtherType 88-8E), MACsec frames (EtherType 88-E5), and MAC Control frames (EtherType 88-08) are permitted and 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 through the CLI command of "mka-policy confidentiality-offset command". An offset value of 0 does not offset the encryption and offset values of 30 and 50 offset the encryption by 30 and 50 characters respectively. An Integrity Check Value (ICV) that is 16 bytes in length is derived with the Secure Association Key (SAK) and is used to provide assurance of the integrity of MPDUs. The TOE derives the ICV from a CAK using KDF, using the SCI as the most significant bits of the IV and the 32 least significant bits of the PN as the IV.

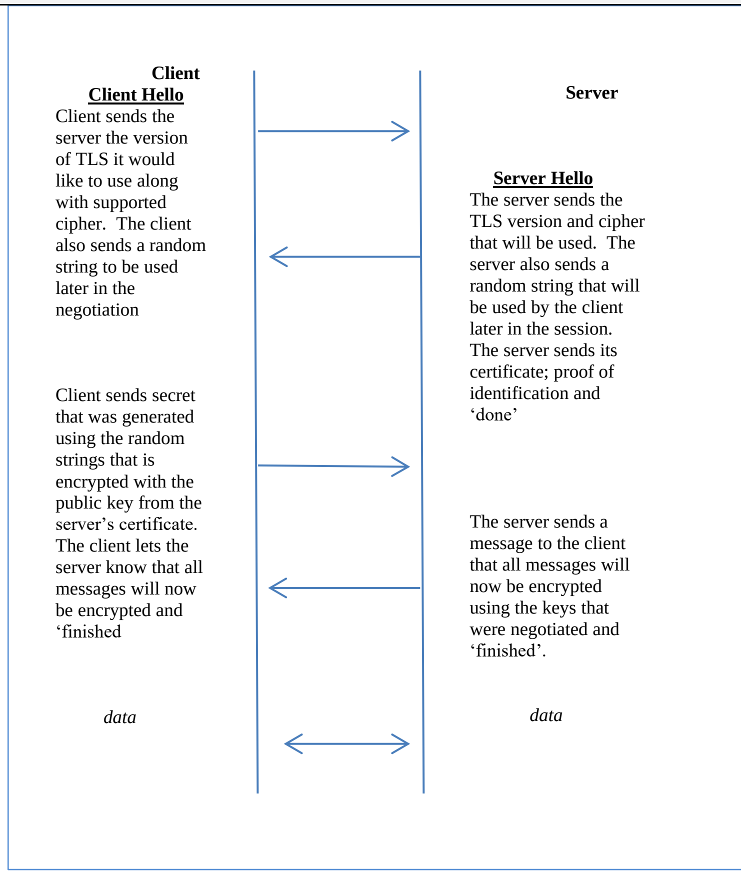
TOE SFRs	How the SFR is Met
FCS_MACSEC_EXT.3	Each SAK is generated using the KDF specified in SP800-108 (KDF Validation System), clause 6.2.1 using the following transform - KS-nonce = a nonce of the same size as the required SAK, obtained from an RNG each time an SAK is generated. The CAK is based on AES cipher in CMAC mode, with key sizes of 128 and 256 bits. 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, which are derived via key derivation function as defined in SP800-108 KDF (CMAC) are tied to the identity of the CAK, and thus restricted to use with that particular CAK. These are the ICV Key (ICK) used to verify the integrity of MPDUs and to prove that the transmitter of the MKPDU possesses the CAK, and the Key Encrypting Key (KEK) used by the Key Server, elected by MKA, to transport a succession of SAKs, for use by MACsec, to the other member(s) of a CA. The size of the key is based on the configured AES key sized used. If using AES 128-bit CMAC mode encryption, the key string will be 32-bit hexadecimal in length. If using 256-bit encryption, the key string will be 64-bit hexadecimal in length. The TOE's random bit generator is used for creating these unique nonces.
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 GCM with a key size of 128 or 256 bits in accordance with AES as specified in ISO 18033-3, AES Key Wrap in CMAC mode as specified in NIST SP 800-38F, and GCM as specified in ISO 19772. The "Key-chain macsec lifetime" key configuration command is used to specify the lifetime for CAKs. The "MACSEC Key-chain key" configuration command is used to specify the length of the CKN that is allowed to be between 1 and 32 octets.
FCS_MKA_EXT.1	The TOE implements Key Agreement Protocol (MKA) in accordance with IEEE 802.1X-2010 and 802.1Xbx-2014. The 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 PDUs. The Delay protection does not operate if and when 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 MKPDUs that do not satisfy the requirements listed under FCS_MKA_EXT.1.8 in Section 5.3.2.15. 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

<table>
<tr><th>TOE SFRs</th><th>How the SFR is Met</th></tr>
<tr><td></td><td>exchange, the MKA ICV is validated with a Connectivity Association Key (CAK), which is effectively a secret key.

For the Data Integrity Check, MACsec uses MKA to generate an Integrity Check Value (ICV) for the frame arriving on the port. If the generated ICV is the same as the ICV in the frame, then the frame is accepted; otherwise, it is dropped. The key string is the Connectivity Association Key (CAK) that is used for ICV validation by the MKA protocol.

The Key Server distributes a SAK by pairwise CAKs.</td></tr>
<tr><td>FCS_SSHS_EXT.1</td><td>SSHv2 is implemented according to the following RFCs: 4251, 4252, 4253, 4254, 4344 and 6668. The TOE implementation of SSHv2 supports the following:
Public key algorithms for authentication: RSA Signature Verification.
The TOE allows public key based authentication for remote administrative users: SSH client’s presented public key matches one that is stored within the TOE’s authorized_keys file.
Local password-based authentication for administrative users accessing the TOE through SSHv2.
Remote CLI SSHv2 sessions are limited to an administrator configurable session timeout period.
Encryption algorithms, AES128-CTR, AES256-CTR to ensure confidentiality of the session.
The TOE’s implementation of SSHv2 supports hashing algorithm hmac-sha1 to ensure the integrity of the session.
The TOE’s implementation of SSHv2 can be configured to only allow Diffie-Hellman Group 14 (2048-bit keys) Key Establishment, as required by the PP.
Packets greater than 65,535 bytes in an SSH transport connection are dropped. Large packets are detected by the SSH implementation, and dropped internal to the SSH process.
The TOE can also be configured to ensure that SSH re-key of no longer than one hour and no more than one gigabyte of transmitted data for the session key. Rekeying is performed upon reaching the threshold that is hit first.
Please refer to Table 5 for all the CAVP references.</td></tr>
<tr><td>FCS_TLSC_EXT.1</td><td>The TOE supports TLS 1.2 to protect the sessions to the remote audit server.
TLS is also used to protect the TLS sessions with the TOE, which supports the mandatory ciphersuite as well as the following optional ciphersuite:
TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268
TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268
The TOE does not support NIST Curves in the TLS Client Hello.

The TOE will only establish a connection if the peer presents a valid certificate during the handshake.
Where the TOE is the client, such as connecting to the remote syslog server, the handshake above is the same process except the server (remote syslog server) would not request the client certificate in the Server Hello, see the following:</td></tr>
</table>

TOE SFRs	How the SFR is Met
	 Any session where the server offers the following in the server hello: SSL 2.0, SSL 3.0, TLS 1.0 and TLS 1.1 will be rejected by the TOE (client). Using IP addresses and wildcards is not supported in identity certificates. Certificate pinning is not supported. The TOE requires Subject Alternative Names (SANs) "the reference identifiers" for a successful connection. SANs contain one or more alternate names and uses any variety of name forms for the entity that is bound by the Certificate Authority (CA) to the certified public key. These alternate names are called "Subject Alternative Names" (SANs). Possible names include: • DNS name
FIA_AFL.1	The TOE provides the privileged administrator the ability to specify the maximum number of unsuccessful authentication attempts before privileged administrator or non-privileged administrator is locked out through the administrative CLI using a privileged CLI command. While the TOE supports a range from 1-24, in the evaluated configuration, the maximum number of failed attempts is recommended to be set to 3. Once the remote user is locked out, their account will not be accessible until the configured timer for lockout has been exceeded. Once the lockout time is over, then the administrator user can attempt to login again. At no point is administrator access completely unavailable when remote administrators are locked out due to

TOE SFRs	How the SFR is Met
	unsuccessful password attempts. Local console access is always available. Administrator lockouts are not applicable to the local console.
FIA_PMG_EXT.1	The TOE supports the local definition of users with corresponding passwords. The passwords can be composed of any combination of upper and lower case letters, numbers, and special characters (that include: “!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, and “)”. Minimum password length is settable by the Authorized Administrator, and can be configured for minimum password lengths of 15 characters or greater.
FIA_PSK_EXT.1	The TOE supports use of pre-shared keys for MACsec key agreement protocols. 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.”
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 except for the login warning banner that is displayed prior to user authentication. Administrative access to the TOE is facilitated through the TOE’s CLI. The TOE mediates all administrative actions through the CLI. Once a potential administrative user attempts to access the CLI of the TOE through either a directly connected console or remotely through an SSHv2 secured connection, the TOE prompts the user for a user name and password. Only after the administrative user presents the correct authentication credentials will access to the TOE administrative functionality be granted. No access is allowed to the administrative functionality of the TOE until an administrator is successfully identified and authenticated. The TOE provides a local password based authentication mechanism for authentication of authorized administrators. 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 grant administrative access (if the combination of username and password is correct) or indicate that the login was unsuccessful. The TOE does not provide a reason for failure in the cases of a login failure.
FIA_UAU.7	When a user enters their password at the local console, the TOE displays no characters so that the user password is obscured. For remote session authentication, the TOE does not echo any characters as they are entered.
FIA_X509_EXT.1/Rev FIA_X509_EXT.2 FIA_X509_EXT.3	The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for TLS connections. Public key infrastructure (PKI) credentials, such as private keys and certificates are stored securely. The identification and authentication, and authorization security functions protect an unauthorized user from gaining access to the storage . The certificate request message includes the public key and common name per RFC 2986. The certificate validation checking takes place during the TLS session establishment and at time of import. The TOE conforms to standard RFC 5280 for certificate and path validation (i.e., peer certificate checked for expiration, peer certificate checked if

TOE SFRs	How the SFR is Met
	signed by a trusted CA in the trust chain, peer certificate checked for unauthorized modification, peer certificate checked for revocation). The TOE supports Self-signed certificate enrollment for a trust point to obtain a certificate from a CA: The certificate chain establishes a sequence of trusted certificates, from a peer certificate to the root CA certificate. Within the PKI hierarchy, all enrolled peers can validate the certificate of one another if the peers share a trusted root CA certificate or a common subordinate CA. Each CA corresponds to a trust point. When a certificate chain is received from a peer, the default processing of a certificate chain path continues until the first trusted certificate, or trust point, is reached. Both the certificate request message and the certificates themselves provide protection in that they are digitally signed. If a certificate is modified in any way, it would be invalidated. The digital signature verifications process would show that the certificate had been tampered with when the hash value would be invalid. Checking is also done for the basicConstraints extension and the CA flag to determine whether they are present and set to TRUE. The local certificate that was imported must contain the basic constraints extension with the CA flag set to true, the check also ensure that the key usage extension is present, and the keyEncipherment bit or the keyAgreement bit or both are set. If they are not, the certificate is not accepted. Only one certificate is imported since the only device is a syslog server, so the TOE chooses this certificate basicConstraints checking is performed at the time of authentication during the connection attempt. If the connection to determine the certificate validity cannot be established, the certificate is not accepted. The administrators can configure a trust chain by importing the CA certificate(s) that signed and issued the server (syslog) certificate. This will tell the TOE which CA certificate(s) to use during the validation process. If the TOE does not find the trusted root CA, the TLS connection to the syslog server will fail. When the TOE is able to contact the CRL distribution point for certificate revocation checking, the TOE will reject the TLS session if the remote trust point's (e.g. syslog server's) certificate has been revoked.
FMT_MOF.1/ManualUpdate	The TOE provides administrative users with a CLI to interact with and manage the security functions of the TOE. 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. Therefore, 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 user accounts and roles, audit data, audit server information, configuration data, security attributes, X.509 certificates, login banners, inactivity timeout values, password complexity setting, TOE updates and session thresholds via the CLI. The TOE restricts the access to manage TSF data that can affect security functions of the TOE to the Authorized Administrator/Security Administrator roles.
FMT_MOF.1/Services	
FMT_MTD.1/CoreData	

<table>
<tr><th>TOE SFRs</th><th>How the SFR is Met</th></tr>
<tr><td>FMT_MTD.1/CryptoKeys</td><td>Manual software updates can only be done by the authorized administrator through CLI. These updates include software upgrades.

The Security Administrators (a.k.a Authorized Administrators) 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 Security Administrator is able to manage the cryptographic keys (generating keys, importing keys, or deleting keys) that are used in TLS and SSH communications. These keys can be managed via CLI as part of following operations:
• TLS public/private keys – CSR (keypair) generation, certificate import/export, Trust store management
• TLS/SSH session keys– as part of session establishment and termination
• SSH public/private keys – generate keypair, import/export public keys, public key-based authentication
• MACsec keys – as part of MACsec session establishment and termination
• Zeroize – delete keys

See FMT_SMF.1 for services the Security Administrator is able to start and stop. Management functionality of the TOE is provided through the TOE CLI.</td></tr>
<tr><td>FMT_SMF.1</td><td>The TOE provides all the capabilities necessary to securely manage the TOE and the services provided by the TOE. The management functionality of the TOE is provided through the TOE CLI. The specific management capabilities available from the TOE include -
• Local and remote administration of the TOE and the services provided by the TOE via the TOE CLI, as described below;
• The ability to manage the warning banner message and content – 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 manage the time limits of session inactivity which allows the Authorized Administrator the ability to set and modify the inactivity time threshold.
• The ability to update the IOS-XR software. The validity of the image is provided using digital signature or hash comparison prior to installing the update
• The ability to manage audit behavior and the audit logs which allows the Authorized Administrator to configure the audit logs, view the audit logs, and to clear the audit logs
• The ability to display the log on banner and to allow any network packets as configured by the authorized administrator may flow through the router prior to the identification and authentication process
• 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 authentication failure parameters for FIA_AFL.1.
• The ability to configure the cryptographic functionality.
• The ability to configure thresholds for SSH rekeying.
• The ability to set the time which is used for time-stamps.</td></tr>
</table>

TOE SFRs	How the SFR is Met
	• The ability to configure the reference identifier for the peer. • Ability to import X.509v3 certificates to the TOE's trusted store. • Ability to manage the trusted public keys database. • Ability to configure the time interval for administrator lockout due to excessive authentication failures. The ability of the Security Administrator to: • 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 command] • Configure the number of failed administrator authentication attempts that will cause an account to be locked out • Ability to start and stop services
FMT_SMR.2	The TOE platform maintains both privileged and semi-privileged administrator roles. The terms "Authorized Administrator" and "Security Administrator" are used interchangeable 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 assigned role determines the functions the user can perform, hence the authorized administrator with the appropriate privileges. The TOE supports both local administration via a directly connected console cable and remote administration via SSH.
FPT_CAK_EXT.1	During the setup and configuration of the TOE and the MACsec functionality, the Authorized Administrator issues the command – "password encryption aes". This prevents the CAK value from being shown in clear text to the administrators on the CLI when the "show run" output is displayed. In addition, CAK data is stored in secure directory that is not readily accessible to administrators.
FPT_FLS.1/SelfTest	Whenever a failure occurs within the TOE that results in the TOE ceasing operation, the TOE securely disables its interfaces to prevent the unintentional flow of any information to or from the TOE. The TOE shuts down by reloading and will continue to reload as long as the failures persist. This functionally prevents any failure of power-on self-tests, failure of integrity check of the TSF executable image, failure of noise source health tests from causing an unauthorized information flow. There are no failures that circumvent this protection.
FPT_RPL.1	MPDUs are replay protected in the TOE. Also, the MKA frames are guarded against replay (If a MKPDU with duplicate MN (member number) and not latest MN comes along, then this MKPDU will be dropped and not processed further). Replayed data is discarded and logged by the TOE.

TOE SFRs	How the SFR is Met
FPT_SKP_EXT.1 FPT_APW_EXT.1	The TOE stores all private keys in a secure directory that is not readily accessible to administrators. All pre-shared and symmetric keys are stored in a hashed format that are non-readable, hence no interface access. All passwords are obscured via hashing in a secure directory. The passwords are non-readable and encrypted. In this manner, the TOE ensures that plaintext user passwords will not be disclosed even to administrators. This is provided by default.
FPT_STM_EXT.1	The TOE provides a source of date and time information used in audit event timestamps. The clock function is reliant on the system clock. This date and time is used as the time stamp that is applied to TOE generated audit records and used to track inactivity of administrative sessions.
FPT_TUD_EXT.1	An Authorized Administrator can query the software version running on the TOE and can initiate updates to software images. When software updates are made available by Cisco, an administrator can obtain, verify the integrity of, and install those updates. The updates can be downloaded from the software.cisco.com. The cryptographic hashes (i.e., public hashes/SHA-512) are used to verify software/firmware update files (to ensure they have not been modified from the originals distributed by Cisco) before they are used to update the applicable TOE components. Once the file is downloaded from the Cisco.com web site, and upon installation of a TOE update, a digital signature verification check will automatically be performed to ensure it has not been modified since distribution. The authorized source for the digitally signed updates is "Cisco Systems, Inc.". The hash value can be displayed by hovering over the software image name under details on the Cisco.com web site. If the hashes do not match, contact Cisco Technical Assistance Center (TAC). The TOE image files are digitally signed so their integrity can be verified during the boot process, and an image that fails an integrity check will not be loaded. The digital certificates used by the update verification mechanism are contained on the TOE. Detailed instructions for how to verify the hash value are provided in the administrator guidance for this evaluation.
FPT_TST_EXT.1	The TOE is designed to runs the suite of power-on self-tests that comply with the FIPS140-2 requirements for self-test (eg know answer tests (KATs) and zeroization tests), during initial start-up to verify its correct operation. If any of the tests fail the security administrator will have to log into the CLI to determine which test failed and why. If the tests pass successfully the router will continue bootup and normal operation. During the system bootup process (power on or reboot), all the Power on Startup Test (POST) components for all the cryptographic modules perform the POST for the corresponding component (hardware or software). These tests include: • AES Known Answer Test – For the encrypt test, a known key is used to encrypt a known plain text value resulting in an encrypted value. This encrypted value is compared to a known encrypted value to ensure that the encrypt operation is working correctly. The decrypt test is just the opposite. In this test a known key is used to decrypt a known

TOE SFRs	How the SFR is Met
	encrypted value. The resulting plaintext value is compared to a known plaintext value to ensure that the decrypt operation is working correctly. • RSA Signature Known Answer Test (both signature/verification) This test takes a known plaintext value and Private/Public key pair and used the public key to encrypt the data. This value is compared to a known encrypted value to verify that encrypt operation is working properly. The encrypted data is then decrypted using the private key. This value is compared to the original plaintext value to ensure the decrypt operation is working properly. • RNG/DRBG Known Answer Test – For this test, known seed values are provided to the DRBG implementation. The DRBG uses these values to generate random bits. These random bits are compared to known random bits to ensure that the DRBG is operating correctly. • HMAC Known Answer Test – For each of the hash values listed, the HMAC implementation is fed known plaintext data and a known key. These values are used to generate a MAC. This MAC is compared to a known MAC to verify that the HMAC and hash operations are operating correctly. • SHA-1/256 Known Answer Test – For each of the values listed, the SHA implementation is fed known data and key. These values are used to generate a hash. This hash is compared to a known value to verify they match, and the hash operations are operating correctly. • Software Integrity Test – The Software Integrity Test is run automatically whenever the IOS system images is loaded and confirms that the image file that's about to be loaded has maintained its integrity. If any component reports failure for the POST, the system crashes and appropriate information is displayed on the screen and saved in the crashinfo file. All ports are blocked from moving to forwarding state during the POST. If all components of all modules pass the POST, the system is placed in FIPS PASS state and ports are allowed to forward data traffic. These tests are sufficient to verify that the correct version of the TOE software is running as well as that the cryptographic operations are all performing as expected.
FTA_SSL_EXT.1 FTA_SSL.3	An administrator can configure maximum inactivity times individually for both local and remote administrative sessions through the use of the "session-timeout" setting applied to the console. When a session is inactive (i.e., no session input from the administrator) for the configured period of time the TOE will terminate the session, and no further activity is allowed requiring the administrator to log in (be successfully identified and authenticated) again to establish a new session. If a remote user session is inactive for a configured period of time, the session will be terminated and will require authentication to establish a new session. The allowable inactivity timeout range is from 1 to 65535 seconds. Administratively configurable timeouts are also available for the EXEC level access (access above level 1) through use of the "exec-timeout" setting.

TOE SFRs	How the SFR is Met
FTA_SSL.4	An administrator is able to exit out of both local and remote administrative sessions. Each administrator logged onto the TOE can manually terminate their session using the “exit” command.
FTA_TAB.1	The TOE displays a privileged Administrator specified banner on the CLI management interface prior to allowing any administrative access to the TOE. This interface is applicable for both local (via console) and remote (via SSH) TOE administration.
FTP_ITC.1	The TOE protects communications with the external audit server using TLS to secure the communications channel. TLS uses the keyed hash as defined in FCS_COP.1/KeyedHash and cryptographic hashing functions FCS_COP.1/Hash. This protects the data from modification of data by hashing that verify that data has not been modified in transit. In addition, encryption of the data as defined in FCS_COP.1/DataEncryption is provided to ensure the data is not disclosed in transit. The TOE protects communications between the TOE and the remote audit server using TLS. This provides a secure channel to transmit the log events. MACsec is also used to secure communication channels between MACsec peers at Layer 2.
FTP_TRP.1 /Admin	All remote administrative communications take place over a secure encrypted SSHv2 session. The SSHv2 session is encrypted using AES encryption. The remote users are able to initiate SSHv2 communications with the TOE.

7 Annex A: Key Zeroization

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

Table 19 TOE Key Zeroization

Name	Description of Key	Zeroization
Diffie-Hellman Shared Secret	This is the shared secret used as part of the Diffie-Hellman key exchange. This key is stored in DRAM.	Automatically after completion of DH exchange. Overwritten with: 0x00
Diffie Hellman private exponent	This is the private exponent used as part of the Diffie-Hellman key exchange. This key is stored in DRAM.	Zeroized upon completion of DH exchange. Overwritten with: 0x00
MACsec Security Association Key (SAK)	The SAK is used to secure the control plane traffic. This key is stored in internal ASIC register.	Automatically when MACsec session terminated. Overwritten with: 0x00
MACsec Connectivity Association Key (CAK)	The CAK secures the control plane traffic. This key is stored in internal ASIC register.	Automatically when MACsec session terminated. Overwritten with: 0x00
MACsec Key Encryption Key (KEK)	The Key Encrypting Key (KEK) is used by Key Server, elected by MKA, to transport a succession of SAKs, for use by MACsec, to the other member(s) of a Secure Connectivity Association (CA). This key is stored in internal ASIC register.	Automatically when MACsec session terminated. Overwritten with: 0x00
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. Overwritten with: 0x00
SSH Private Key	Once the function has completed the operations requiring the RSA key object, the module over writes the entire object (no matter its contents) using memset. This overwrites the key with all 0's. This key is stored in NVRAM.	Zeroized using the following command: **# crypto key zeroize rsa** Overwritten with: 0x00
SSH Session Key	Once the function has completed the operations requiring the RSA key object, the module over writes the entire object (no matter its contents). This is called by the ssh_close function when a session is ended. This key is stored in DRAM.	Automatically when the SSH session is terminated. Overwritten with: 0x00
User Password	This is a variable 15+ character password that is used to authenticate local users. The password is stored in NVRAM.	Zeroized by overwriting with new password
Enable Password (if used)	This is a variable 15+ character password that is used to authenticate local users at a higher privilege level. The password is stored in NVRAM.	Zeroized by overwriting with new password
RNG Seed	This seed is for the RNG. The seed is stored in DRAM.	Zeroized upon power cycle the device
RNG Seed Key	This is the seed key for the RNG. The seed key is stored in DRAM.	Zeroized upon power cycle the device

Name	Description of Key	Zeroization
AES Key	The results are zeroized by overwriting the values with 0x00. This is called by the ssh_close function when a session is ended. This key is stored in DRAM	Automatically when the SSH/TLS session is terminated. Overwritten with: 0x00
TLS server private key	This key is used for authentication, so the server can prove who it is. The private key used for SSLv3.1/TLS secure connections. The key is stored in NVRAM.	Zeroized by overwriting with new key
TLS server public key	This key is used to encrypt the data that is used to compute the secret key. The public key used for SSLv3.1/TLS secure connection. The key is stored in NVRAM.	Zeroized by overwriting with new key
TLS pre-master secret	The pre-master secret is the client and server exchange of random numbers and a special number, the pre-master secret, This pre-master secret is using asymmetric cryptography from which new TLS session keys can be created. The key is stored in SDRAM.	Automatically after TLS session terminated. The value is overwritten with “0x00.”
TLS session encryption key	The session encryption key is unique for each session and is based on the shared secrets that were negotiated at the start of the session. The Key is used to encrypt TLS session data. The key is stored in SDRAM.	Automatically after TLS session terminated. The value is overwritten with “0x00.”
TLS session integrity key	This key is used to provide the privacy and TLS data integrity protection. The key is stored in SDRAM.	Automatically after TLS session terminated. The entire object is overwritten with zeros

8 Annex B: NIAP Technical Decisions (TD)

The following Technical Decisions apply to the NDcPPv2.0e and MACsecEPv1.2:

Table 20 NIAP Technical Decisions

TD Identifier	TD Name	Protection Profiles	References	Publication Date	Applicable?
TD0670	NIT Technical Decision for Mutual and Non-Mutual Auth TLSC Testing	CPP_ND_V2.2E	ND SD2.2, FCS_TLSC_EXT.2.1	2022.09.16	No, SFR 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	2022.08.30	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	ND SD2.2, FCS_SSHC_EXT.1	2022.03.21	No, SFR not claimed
TD0635	NIT Technical Decision for TLS Server and Key Agreement Parameters	CPP_ND_V2.2E	FCS_TLSS_EXT.1.3, NDSD v2.2	2022.03.21	Yes
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	Yes
TD0633	NIT Technical Decision for IPsec IKE/SA Lifetimes Tolerance	CPP_ND_V2.2E	ND SD2.2, FCS_IPSEC_EXT.1.7, FCS_IPSEC_EXT.1.8	2022.03.21	No, SFR not claimed
TD0632	NIT Technical Decision for Consistency with Time Data for vNDs	CPP_ND_V2.2E	ND SD2.2, FPT_STM_EXT.1.2	2022.03.21	No, vND is not claimed
TD0631	NIT Technical Decision for Clarification of public key authentication for SSH Server	CPP_ND_V2.2E	ND SDv2.2, FCS_SSHS_EXT.1, FMT_SMF.1	2022.03.21	Yes
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_FUNCTIONALITY, ACRONYMS	2021.05.21	No, vND is not claimed
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

TD Identifier	TD Name	Protection Profiles	References	Publication Date	Applicable?
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	Yes
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
TD0538	The NIT has issued a technical decision for Outdated link to allowed-with list	CPP_ND_V2.1, CPP_ND_V2.2E	Section 2	2020.07.13	Yes
TD0537	The NIT has issued a technical decision for Incorrect reference to FCS_TLSC_EXT.2.3	CPP_ND_V2.2E	FIA_X509_EXT.2.2	2020.07.13	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
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

TD Identifier	TD Name	Protection Profiles	References	Publication Date	Applicable?
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	Yes

9 Annex C: References

The following documentation was used to prepare this ST:

Table 21 References

Identifier	Description
[CC_PART1]	Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and general model, dated April 2017, version 3.1, Revision 5
[CC_PART2]	Common Criteria for Information Technology Security Evaluation – Part 2: Security functional components, dated April 2017, version 3.1, Revision 5
[CC_PART3]	Common Criteria for Information Technology Security Evaluation – Part 3: Security assurance components, April 2017, version 3.1, Revision 5
[CEM]	Common Methodology for Information Technology Security Evaluation – Evaluation Methodology, April 2017, version 3.1, Revision 5
[NDcPP]	collaborative Protection Profile for Network Devices, Version 2.2e, 23 March 2020
[MACSECEP]	Network Device Protection Profile Extended Package MACsec Ethernet Encryption (MACSECEP), Version 1.2
[800-38A]	NIST Special Publication 800-38A Recommendation for Block 2001 Edition Recommendation for Block Cipher Modes of Operation Methods and Techniques December 2001
[800-56A]	NIST Special Publication 800-56A, March 2007 Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography (Revised)
[800-56B]	NIST Special Publication 800-56B Recommendation for Pair-Wise, August 2009 Key Establishment Schemes Using Integer Factorization Cryptography
[FIPS 140-2]	FIPS PUB 140-2 Federal Information Processing Standards Publication Security Requirements for Cryptographic Modules May 25, 2001
[FIPS PUB 186-3]	FIPS PUB 186-3 Federal Information Processing Standards Publication Digital Signature Standard (DSS) June, 2009
[FIPS PUB 186-4]	FIPS PUB 186-4 Federal Information Processing Standards Publication Digital Signature Standard (DSS) July 2013
[FIPS PUB 198-1]	Federal Information Processing Standards Publication The Keyed-Hash Message Authentication Code (HMAC) July 2008
[NIST SP 800-90A Rev 1]	NIST Special Publication 800-90A Recommendation for Random Number Generation Using Deterministic Random Bit Generators January 2015
[FIPS PUB 180-3]	FIPS PUB 180-3 Federal Information Processing Standards Publication Secure Hash Standard (SHS) October 2008