Network Device Collaborative Protection Profile (NDcPP)
Extended Package
MACsec Ethernet Encryption

March 04 2016
Version 1.1

Table of Contents

Abbreviations and acronyms

AES	Advanced Encryption Standard
AN	Association Number
CA	Secure Connectivity Association
CAK	Secure Connectivity Association Key
CKN	Secure Connectivity Association Key Name
CMAC	Cipher-based Message Authentication Code
CRC	Cyclic Redundancy Check
DA	Destination Address
EAP	Extensible Authentication Protocol
EAP-TLS	EAP Transport Layer Security
EAPOL	EAP over LANs
GCM	Galois Counter Mode
ICK	ICV Key
ICV	Integrity Check Value
IP	Internet Protocol
IV	Initialization Vector
KaY	MAC Security Key Agreement Entity
KDF	Key Derivation Function
KEK	Key Encrypting Key
KI	Key Identifier
MAC	Media Access Control
MKA	MACsec Key Agreement protocol
MKPDU	MACsec Key Agreement Protocol Data Unit
MPDU	MAC Protocol Data Unit
MSAP	MAC Service Access Point
MSDU	MAC Service Data Unit
MSK	Master Session Key
NID	Network Identity
NIST	National Institute of Standards and Technology
PAC	Port Access Controller
PACP	Port Access Control Protocol
PAE	Port Access Entity
PDU	Protocol data unit
PN	Packet Number
PSK	pre-shared key
RADIUS	Remote Authentication Dial in User Service
RNG	Random number generator
SA	Secure Association
SAI	Secure Association Identifier
SAK	Secure Association Key
SC	Secure Channel
SCI	Secure Channel Identifier
SecTAG	MAC Security TAG
SecY	MAC Security Entity
SCI	Secure Channel Identifier
SecTAG	MAC Security TAG
SecY	MAC Security Entity
SNMP	Simple Network Management Protocol

1 Introduction

This Extended Package (EP) describes security requirements for a network device that implements Media Access Control Security (MACsec) encryption to secure communications over a trusted channel and is intended to provide a minimal, baseline set of requirements that are targeted at mitigating well defined and described threats. However, this EP is not complete in itself, but rather extends the *Security Requirements for Network Devices collaborative Protection Profile* (NDcPP). This introduction will describe the features of a compliant Target of Evaluation (TOE), and will also discuss how this EP is to be used in conjunction with the NDcPP.

1.1 Conformance Claims

The collaborative Protection Profile for Network Devices (NDcPP) defines the baseline Security Functional Requirements (SFRs) and Security Assurance Requirements (SARs) for network infrastructure devices in general. This EP serves to extend the NDcPP baseline with additional SFRs and associated 'Assurance Activities' specific to Media Access Control Security (MACsec) devices. Assurance Activities are the actions that the evaluator performs in order to determine a TOE's compliance to the SFRs. This EP conforms to Common Criteria for Information Technology Security Evaluation, Version 3.1, and Revision 4. It is CC Part 2 extended and CC Part 3 conformant.

1.2 How to Use This Extended Package

As an EP of the NDcPP, it is expected that the content of both this EP and the NDcPP be appropriately combined in the context of each product-specific Security Target (ST). This EP has been specifically defined so that it is possible to define a Target of Evaluation (TOE) that contains the security functional requirements (SFRs) of both the NDcPP and this EP without contradictions or ambiguities. An ST must identify the applicable versions of the NDcPP (see http://www.niap-ccevs.org/pp/ for the current version) and this EP in its conformance claims.

1.3 Compliant Targets of Evaluations

This EP specifically addresses MACsec, which allows authorized systems using Ethernet Transport to maintain confidentiality of transmitted data and to take measures against frames that are transmitted or modified by unauthorized devices.

MACsec protects communication between trusted components of the network infrastructure, thus protecting the network operation. It facilitates maintenance of correct network connectivity and services as well as isolation of denial of service attacks.

The hardware, firmware, and software of the MACsec device define the physical boundary. All of the security functionality is contained and executed within the physical boundary of the device. For example, given a computer with an Ethernet card, the whole computer is considered to be within the boundary.

Since this EP builds on the NDcPP, conformant TOEs are obligated to implement the functionality required in the NDcPP along with the additional functionality defined in this EP in response to the threat environment discussed later in this document.

It is intended that the set of requirements in this EP is limited in scope in order to promote quicker, less costly evaluations that provide some value to end users.

1.4 Deployment Scenario

A pair of MACsec devices connected by a physical medium can protect Ethernet frames switched or routed from one device to the other. The two MACsec devices are provided with a Connectivity Association Key (CAK) and utilize the MACsec Key Agreement (MKA) protocol to create a secure tunnel. MKA is used by the two MACsec devices to agree upon MACsec keys. A policy should be installed to protect traffic between the devices, with the exception of the MKA or Ethernet control traffic such as EAP over LAN (EAPOL) frames.

In a point-to-point (P2P) deployment, two devices will protect traffic originating in protected networks traversing an untrusted link between them or traffic that is contained within an internal network but requires additional security that is provided by end-to-end encryption. The devices will first exchange MKA frames, which serve to determine the peer is an authorized peer, and agree upon a shared key and MACsec cipher suite used to set up a transmit (Tx) Security Association (SA) and a receive (Rx) SA. Once the SAs are set up, MACsec-protected frames traverse the unprotected link.

This scenario could also be for a LAN or P2P Ethernet Virtual Private Line services. Ethernet private line (EPL) and Ethernet virtual private line (EVPL) are carrier Ethernet data services defined by the Metro Ethernet Forum. EPL provides a point-to-point Ethernet virtual connection (EVC) between a pair of dedicated user–network interfaces (UNIs), with a high degree of transparency. EVPL provides a point-to-point or point-to-multipoint connection between a pair of UNIs.   A difference between the EVPL and EPL is the degree of transparency - while EPL is highly transparent, filtering only the pause frames, EVPL is required to either peer or drop most of the Layer 2 Control Protocols.

2 Security Problem Description

The MACsec device is a specialized type of network device that provides security for Ethernet traffic. This type of product is intended to provide security functions that are related to the configuration and implementation of MACsec communications. Since the use of encryption implies that a MACsec device is either connected to an untrusted network or transmitting data in an internal network that requires more security than what is provided by that network, it is necessary for trusted communications channels and secure administration to be implemented in order to address threat vectors that originate from untrusted sources.

This EP details the functional requirements and threats specific to a network device that performs MACsec. Additional functional requirements pertaining to the general network device capability of a MACsec product are specified in the NDcPP and are not repeated here.

2.1 Threats

As an extension to the Network Device cPP, a MACsec product will face the same threats that apply generally to all network devices. However, due to the specialized security features of MACsec, some of these threats are applicable in a more specific context.

2.1.1 Inappropriate Access to Services

A MACsec device may sit on the periphery of a network, which means that it may have an externally-facing interface to a public network. Devices located in the public network may attempt to exercise services located on the internal network that are intended to be accessed only from within the internal network or externally accessible only from specifically authorized devices. If the MACsec device allows unauthorized external devices access to the internal network, these devices on the internal network may be subject to compromise. Similarly, if two MACsec devices are deployed to facilitate end-to-end encryption of traffic that is contained within a single network, an attacker could use an insecure MACsec device as a method to access devices on a specific segment of that network such as an individual LAN.

T.NETWORK_ACCESS An attacker may send traffic through the TOE that enables them to access devices in the TOE's Operational Environment without authorization.

2.1.2 Untrusted Communication Channels

A generic network device may be threatened by the use of insecure communications channels to transmit sensitive data. The attack surface of a MACsec device also includes the MACsec trusted channels. Inability to secure communications channels, or failure to do so correctly, would expose user data that is assumed to be secure to the threat of unauthorized disclosure.

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.

2.1.3 Compromise of Data Integrity

Devices on a network may be exposed to attacks that attempt to corrupt or modify data in transit without authorization. If malicious devices are able to modify and replay data that is transmitted over a trusted channel, then the data contained within the communications may be susceptible to a loss of integrity.

T.DATA_INTEGRITY An attacker may modify data transmitted over the MACsec channel in a way that is not detected by the recipient.

2.2 Assumptions

The assumptions defined for the MACsec device's Operational Environment are identical to those defined by the NDcPP, with the following exception:

The A.NO_THRU_TRAFFIC_PROTECTION assumption defined in the NDcPP does not apply to this EP. A MACsec device is expected to handle traffic between two endpoints by way of a trusted channel between itself and a second MACsec device. MACsec devices are expected to apply port filtering rules to provide rudimentary protection against unauthorized through-traffic.

Security Objectives

3.0 Security Objectives for the TOE

The Security Problem described in Section 2 will be addressed by a combination of cryptographic capabilities. Compliant TOEs will provide security functionality that addresses threats to the TOE and enforces policies that are imposed by law or regulation. The following subsections provide a description of the security objectives required to meet the threats/policies previously discussed.

Note: Specific security objectives are identified (highlighted by *O.*) in each subsection below and are matched with the associated Security Functional Requirements (SFRs) that provide the mechanisms to satisfy the objectives. These include SFRs defined specifically for this EP (see Section 3.2.2) as well as SFRs from the base NDcPP that are either refined in this EP or were optional in the base NDcPP but are mandatory for this EP (see Section 3.2.1).

3.1.1 Data Encryption and Decryption

To address the issues associated with unauthorized modification and disclosure of information, compliant TOEs will implement cryptographic capabilities. These capabilities are intended to maintain confidentiality and allow for detection and modification of data that is transmitted outside of the TOE.

(O.CRYPTOGRAPHIC_FUNCTIONS -> FCS_COP.1(1), FCS_MACSEC_EXT.2, FCS_MACSEC_EXT.3, FTP_ITC.1, FTP_TRP.1)

3.1.2 Authentication

To further address the issues associated with unauthorized disclosure of information, a compliant TOE's authentication ability (MKA) will allow a MACsec peer to establish connectivity associations (CA) with another MACsec peer. MACsec endpoints authenticate each other to ensure they are communicating with an authorized SecY entity (SeY).

(O.AUTHENTICATION -> FCS_MACSEC_EXT.4, FCS_MKA_EXT.1, FIA_PSK_EXT.1)

3.1.3 Port-Based Filtering

To further address the issues associated with unauthorized network access, a compliant TOE's port filtering capability will restrict the flow of network traffic through the TOE based on source address/port and whether or not the traffic represents valid MACsec frames and MACsec Key Agreement Protocol Data Unit(MKPDU)s.

(O.PORT_FILTERING -> FCS_MACSEC_EXT.1, FCS_EAP-TLS_EXT.1 (selection-based), FCS_DEVID_EXT.1 (selection-based), FIA_PSK_EXT.1)

3.1.4 System Monitoring

To address the issues of administrators being able to monitor the operations of the MACsec device, compliant TOEs will implement the ability to log the flow of Ethernet traffic. Specifically, the TOE will provide the means for administrators to configure rules to 'log' when Ethernet traffic grants or restricts access. As a result, the 'log' will result in informative event logs whenever a match occurs. In addition, the establishment of security CAs is auditable, not only between MACsec devices, but also with MAC Security Key Agreement Entities (KaYs).

(O.SYSTEM_MONITORING -> FAU_GEN.1)

3.1.5 Authorized Administration

All network devices are expected to provide services that allow the security functionality of the device to be managed. The MACsec device, as a specific type of network device, has a refined set of management functions to address its specialized behavior. In order to further mitigate the threat of a compromise of its security functionality, the MACsec device prescribes the ability to limit brute-force authentication attempts by enforcing lockout of accounts that experience excessive failures and by limiting access to security-relevant data that administrators do not need to view.

(O.AUTHORIZED_ADMINISTRATION -> FIA_AFL.1, FMT_SNMP_EXT.1 (selection-based), FMT_SMF.1, FPT_CAK_EXT.1, FTP_TRP.1)

3.1.6 TSF Integrity

To mitigate the security risk that the MACsec device may fail during startup, it is required to shut down in the event that any self-test failures occur during startup. This ensures that the device will only operate when it is in a known state.

(O.TSF_INTEGRITY -> FPT_FLS.1(2)/SelfTest)

3.1.7 Replay Detection

A MACsec device is expected to help mitigate the threat of MACsec data integrity violations by providing a mechanism to detect and discard replayed traffic for MACsec protocol data units (MPDUs).

(O.REPLAY_DETECTION -> FPT_RPL.1)

3.1.8 Verifiable Updates

To ensure the authenticity and integrity of software/firmware updates that are loaded onto the MACsec device, it is necessary to provide a mechanism for validating these updates prior to application. The NDcPP provides methods of update verification; this EP specifically requires that a signature-based mechanism be used at minimum.

(O.VERIFIABLE_UPDATES -> FPT_TUD_EXT.1)

3.2 Security Objectives for the Operational Environment

The security objectives for the operational environment for this EP are the same as the security objectives for the operational environment of the base NDcPP with the exception of OE.NO_THRU_TRAFFIC_PROTECTION, which is excluded from this EP. MACsec devices are expected to provide rudimentary through-traffic protection.

4.0 Security Functional Requirements

The Security Functional Requirements (SFRs) included in this section are derived from Part 2 of the *Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 4*, with additional extended functional components.

4.1 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*;
- Refinement made by EP author: Indicated with **bold text**;
- Selection: Indicated with <u>underlined text</u>;
- Assignment within a Selection: Indicated with <u>*italicized and underlined text*</u>;
- Iteration: Indicated by appending the iteration number in parenthesis, e.g. (1), (2), (3) and/or a slash and descriptive string following the SFR name, e.g. /SelfTest ; and
- Extended SFRs are identified by having a label 'EXT' after the requirement name for TOE SFRs.

4.2 TOE Security Functional Requirements

Since this EP extends the NDcPP, it is expected that the security functions that are defined in the base PP are inherited by this EP. For those functions that are defined in the NDcPP but are specified in more detail in this EP, the updated SFRs have been listed in Section 3.2.1 below.

4.2.1 NDcPP Security Functional Requirement Direction

This section instructs the ST author on what selections must be made to certain SFRs contained in the NDcPP in order to satisfy the security objectives defined in this EP, or to mitigate a threat in a more specific or restrictive manner than is specified in the base PP.

This instruction describes the element where the mandatory selection has been made. The ST author may complete the remaining selection items as they wish, to ensure specific capabilities or behavior is present in the TOE.

Full assurance activities are not repeated for the requirements in this section; only the additional testing needed to supplement what has already been captured in the NDcPP is included. As the evaluator assesses the ST and TOE against the SFR, it is important for them to verify that the proper selections have been made and the appropriate tests are performed to demonstrate compliance to the requirements.

4.2.1.1 FAU_GEN.1 Audit Data Generation

There are additional auditable events that serve to extend the FAU_GEN.1 SFR found in the NDcPP. The following events should be combined with those of the NDcPP in the context of a conforming Security Target.

The following auditable events are required for this EP:

SFR	Auditable Event	Additional Audit Record Contents
FCS_MACSEC_EXT.1	Session establishment	Secure Channel Identifier (SCI)
FCS_MACSEC_EXT.1.7	Creation of Connectivity Association	Connectivity Association Key Names
FCS_MACSEC_EXT.3.1	Creation and update of Secure Association Key	Creation and update times
FIA_AFL.1	Administrator lockout due to excessive authentication failures	None
FPT_RPL.1	Detected replay attempt	None

Table 1 - Auditable Events

Assurance Activity
The evaluator shall complete the assurance activity for FAU_GEN.1 as described in the NDcPP for the auditable events defined above in addition to the applicable auditable events that are defined in the NDcPP. The evaluator shall also ensure that the administrative actions defined for this EP are appropriately audited.

4.2.1.2 FCS_COP.1(1) Cryptographic Operation (AES Data Encryption/Decryption)

FCS_COP.1.1(1) 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 in CMAC mode as specified in NIST SP 800-38F, GCM as specified in ISO 19772**.

Application Note: This EP mandates the use of GCM for MACsec and AES Key Wrap for key distribution so this SFR has been further refined from the NDcPP.

Application Note: AES-CMAC is a keyed hash function that is used as part of the key derivation function (KDF) that is used for key generation.

Assurance Activity	
TSS	The evaluator shall verify that the TSS describes the supported AES modes that are required for this EP in addition to the ones already required by the NDcPP.
AGD	No additional guidance review activities are required.
Test	The evaluator shall perform testing for AES-GCM as required by the NDcPP. In addition to the tests specified in the NDcPP for this SFR, the evaluator shall perform the following tests: **CMAC Generation Test**

To test the generation capability of AES-CMAC, the evaluator shall provide to the TSF, for each key length-message length-CMAC length tuple (in bytes), a set of 8 arbitrary key-plaintext tuples that will result in the generation of a known MAC value when encrypted. The evaluator will then verify that the correct MAC was generated in each case.

CMAC Verification Test

To test the generation capability of AES-CMAC, the evaluator shall provide to the TSF, for each key length-message length-CMAC length tuple (in bytes), a set of 20 arbitrary key-MAC tuples that will result in the generation of known messages when verified. The evaluator will then verify that the correct message was generated in each case.

The following information should be used by the evaluator to determine the key length-message length-CMAC length tuples that should be tested:

- Key length: values will include the following:
  - 16
  - 32
- Message length: values will include the following:
  - 0 (optional)
  - Largest value supported by the implementation (no greater than 65536)
  - Two values divisible by 16
  - Two values not divisible by 16
- CMAC length
  - Smallest value supported by the implementation (no less than 1)
  - 16
  - Any supported CMAC length between the minimum and maximum values

4.2.1.3 FMT_SMF.1 Specification of Management Functions

There are additional management functions that serve to extend the FMT_SMF.1 SFR found in the NDcPP. The following functions should be combined with those of the NDcPP in the context of a conforming Security Target:

Ability of a Security Administrator to:

- *Generate a PSK and install it in the CAK cache of a device*
- *Manage the Key Server to create, delete, and activate MKA participants [selection: as specified in 802.1X, sections 9.13 and 9.16 (cf. MIB object ieee8021XKayMkaParticipantEntry) and section 12.2 (cf. function createMKA()), [assignment: other management function]]*
- *Specify a lifetime of a CAK*
- *Enable, disable, or delete a PSK in the CAK cache of a device using [selection: the MIB object ieee8021XKayMkaPartActivateControl, [assignment: other management function]]*
- *Cause Key Server to generate a new group CAK (i.e., rekey the CA) using [selection: MIB object ieee8021XKeyCreateNewGroup. [assignment: other management function]]*
- *Configure the number of failed administrator authentication attempts that will cause an account to be locked out*

[selection:

- *Manually unlock a locked administrator account ,*
- *Configure the time interval for administrator lockout due to excessive authentication failures, [*
- *assignment: any additional management functions],*
- *No other management functions]*

Application Note: IEEE 802.1X specifies MIB objects for management functionality but configuration of management functions via other approved methods is acceptable. The ST author should select either the MIB object or provide the function used to achieve this management functionality.

Assurance Activity	
TSS	The evaluator shall verify that the TSS describes the ability of the TOE to provide the management functions defined in this SFR in addition to the management functions required by the base NDcPP.
AGD	The evaluator shall examine the operational guidance to determine that it provides instructions on how to perform each of the management functions defined in this SFR in addition to those required by the base NDcPP.
Test	The evaluator shall set up an environment where the TOE can connect to two other MACsec devices, identified as devices B and C, with the ability of pre-shared keys to be distributed between them. The evaluator shall configure the devices so that the TOE will be elected key server and principal actor, i.e., has highest key server priority. In addition to the tests specified in the NDcPP for this SFR, the evaluator shall follow the relevant operational guidance to perform the tests listed below. Note that if the TOE claims multiple management interfaces, the tests should be performed for each interface that supports the functions. Test 1: The evaluator shall connect to the PAE of the TOE and install a PSK, initiating the LOGON process, and invoking the cacheCAK(…) function (cf. 802.1X, Section 12.1) to place a PSK in the cache. The evaluator shall use the createMKA() function to specify CKN and the PSK itself as CAK. • Repeat this test for both 128-bit and 256-bit key sizes. • Repeat this test for a CKN of valid length (1-32 octets), and observe success. • Repeat this test again for CKN of invalid lengths zero and 33, and observe failure. Test 2: The evaluator will test the ability of the TOE to enable and disable MKA participants using the management function specified in the ST. . The evaluator shall install pre-shared keys in devices B and C, using the PAE management function cacheCAK(…), which also creates corresponding MKA participants. The evaluator shall disable the MKA participant on device C, then observe that the TOE can communicate with B but neither the TOE nor B can communicate with device C. The evaluator shall re-enable the MKA participant of device B and observe that the TOE is now able to communicate with devices B and C. Test 3: The evaluator shall install PSK on all 3 devices with a short lifetime. The evaluator shall disconnect device B from the test network, disable or deactivate the TOE's listing for device B using the management function specified in the ST, wait for the CAK lifetime to expire, and observe that the TOE generates a new CAK for the TOE and device C. The evaluator shall then

	reconnect device B to the test network and show that the TOE will not allow device B to join the new CA even though it possesses the original PSK. The evaluator shall then reactivate the TOE's original listing for device B and observe that the TOE will rekey and B will be able to reconnect with the CA. Test 4: The evaluator shall connect to the PAE of the TOE, set the management function specified in the ST (e.g., set ieee8021XKeyCreateNewGroup to true), and observe that the TOE distributes a new group CAK.

4.2.1.4 FTP_ITC.1 Inter-TSF Trusted Channel

FTP_ITC.1.1 Refinement: The TSF shall be **capable of using [selection: IPsec, SSH, TLS, HTTPS, SNMPv3, MACsec] to** provide a trusted communication channel between itself and **authorized IT entities supporting the following capabilities: audit server, [selection: authentication server, [*assignment: other capabilities*], no other capabilities]** 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.

Application Note: As per Clause 13 of IEEE 802.1AE-2006, SNMPv3 is permitted for management of MACsec devices. MACsec is permitted to secure the communication channel for management as well as data. This SFR has been further refined from the NDcPP to include both SNMPv3 and MACsec.

Application Note: The other elements of the FTP_ITC.1 SFR are unmodified from how they are defined in the NDcPP.

Assurance Activity
The evaluator shall evaluate this SFR in the manner specified in the NDcPP except that SNMPv3 and MACsec communications shall be tested in addition to any other selected protocols. Testing for these protocols is discussed in Section C.1.

4.2.1.5 FTP_TRP.1 Trusted Path

FTP_TRP.1.1 Refinement: The TSF shall be **capable of using [selection: IPsec, SSH, TLS, HTTPS, SNMPv3] to** provide a trusted communication channel between itself **and authorized remote administrators** that provides confidentiality and integrity, that is, logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from disclosure and provides detection of modification of the channel data.

Application Note: The other elements of the FTP_TRP.1 SFR are unmodified from how they are defined in the NDcPP.

Assurance Activity
The evaluator shall evaluate this SFR in the manner specified in the NDcPP except that SNMPv3 communications shall be tested in addition to any selected protocols. Testing for SNMPv3 is discussed in Section C.1.

4.2.2 MACsec Specific Security Functional Requirements

Security functional requirements in the main body of this EP are divided into those that are inherited from the NDcPP and those that are specific to MACsec TOEs. This section contains requirements that must be met by the TOE and are not covered in the base NDcPP.

4.2.2.1 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) and MACsec frames (EtherType 88-E5) and discard others.

Assurance Activity	
TSS	The evaluator shall examine the TSS to verify that it describes the ability of the TSF to implement MACsec in accordance with IEEE 802.1AE-2006. The evaluator shall also determine that the TSS describes the ability of the TSF to derive SCI values from peer MAC address and port data and to reject traffic that does not have a valid SCI. Finally, the evaluator shall check the TSS for an assertion that only EAPOL and MACsec Ethernet frames are accepted by the MACsec interface.
AGD	There are no guidance activities for this SFR.
Test	The evaluator shall perform the following tests: Test 1: The evaluator shall successfully establish a MACsec channel between the TOE and a MACsec-capable peer in the Operational Environment and verify that the TSF logs the communications. The evaluator shall capture the traffic between the TOE and the Operational Environment to determine the SCI that the TOE uses to identify the peer. The evaluator shall then configure a test system to capture traffic between the peer and the TOE to modify the SCI that is used to identify the peer. The evaluator then verifies that the TOE does not reply to this traffic and logs that the traffic was discarded. Test 2: The evaluator shall send Ethernet traffic to the TOE's MAC address that iterates through the full range of supported EtherType values (refer to http://standards.ieee.org/develop/regauth/ethertype/eth.txt) and observes that traffic for all EtherType values is discarded by the TOE except for the traffic which has an EtherType value of 88-8E or 88-E5. Note that there are a large number of EtherType values so the evaluator is encouraged to execute a script that automatically iterates through each value.

4.2.2.2 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 [selection: 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).

Application Note: The length of the ICV is dependent on the ciphersuite used but will not be less than 8 octets or more than 16 octets at the end of the MPDU. The ICV protects the destination and source MAC address parameters, as well as all the fields of the MPDU.

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.

Assurance Activity	
TSS	The evaluator shall examine the TSS to verify that it describes the methods that the TOE implements to provide assurance of MACsec integrity, including the confidentiality offset(s) used, the use of an ICV (including the supported length), and generating the ICV with the SAK, using the SCI as the most significant bits of the IV and the 32 least significant bits of the PN as the IV.
AGD	If any integrity verifications are configurable such as the confidentiality offset(s) used or the mechanism used to derive an ICK, the evaluator shall verify that instructions for performing these functions are documented.
Test	The evaluator shall perform the following tests: Test 1: The evaluator shall transmit MACsec traffic to the TOE from a MACsec-capable peer in the Operational Environment. The evaluator shall verify via packet captures and/or audit logs that the frame bytes after the MACsec Tag values in the received traffic is not obviously predictable. Test 2: The evaluator shall transmit valid MACsec traffic to the TOE from a MACsec-capable peer in the operational environment that is routed through a test system set up as a man-in-the-middle. The evaluator shall use the test system to intercept this traffic to modify one bit in a packet payload before retransmitting to the TOE. The evaluator shall verify that the traffic is discarded due to an integrity failure.

4.2.2.3 FCS_MACSEC_EXT.3 MACsec Randomness

FCS_ MACSEC_EXT.3.1 The TSF shall generate unique Secure Association Keys (SAKs) using [selection: key derivation from Connectivity Association Key (CAK) per section 9.8.1 of IEEE 802.1X-2010, the TOE's random bit generator as specified by FCS_RBG_EXT.1] such that the likelihood of a repeating SAK is no less than 1 in 2 to the power of the size of the generated key.

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

Application Note: FCS_RBG_EXT.1 is defined in the NDcPP that this EP extends so a conformant MACsec TOE will claim this SFR.

Assurance Activity	
TSS	The evaluator shall examine the TSS to verify that it describes the method used to generate SAKs and nonces and that the strength of the CAK and the size of the CAK's key space are provided.
AGD	There are no guidance activities for this SFR.
Test	Testing of the TOE's MACsec capabilities and verification of the DRBG is sufficient to demonstrate that this SFR has been satisfied.

4.2.2.4 FCS_MACSEC_EXT.4 MACsec Key Usage

FCS_MACSEC_EXT.4.1 The TSF shall support peer authentication using pre-shared keys, [selection: EAP-TLS with DevIDs, no other methods].

Application Note: The definition of the peer's CAK as defined by IEEE 802.1X-2010 is synonymous with the peer authentication performed here. If EAP-TLS is selected, the FCS_EAP-TLS_EXT.1 SFR defined in Appendix C.1 must be included in the TSF.

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

Application Note: This requirement applies to the SAKs that are generated by the TOE. They must be wrapped by the AES Key Wrap method specified in NIST SP 800-38F.

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

FCS_MACSEC_EXT.4.4 The TSF shall associate Connectivity Association Key Names (CKNs) with CAKs 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).

Assurance Activity	
TSS	The evaluator shall check the TSS to ensure that it describes how the SAK is wrapped prior to being distributed using the AES implementation specified in this EP.
AGD	If the method(s) of peer authentication is configurable, the evaluator shall verify that the guidance provides instructions on how to configure this. The evaluator shall also verify that the method of specifying a lifetime for CAKs is described.
Test	The evaluator shall perform the following tests: Test 1: For each supported method of peer authentication in FCS_MACSEC_EXT.4.1, the evaluator shall follow the operational guidance to configure the supported method (if applicable). The evaluator shall set up a packet sniffer between the TOE and a MACsec-capable

	peer in the Operational Environment. The evaluator shall then initiate a connection between the TOE and the peer such that authentication occurs and a secure connection is established. The evaluator shall wait 1 minute and then disconnect the TOE from the peer and stop the sniffer. The evaluator shall use the packet captures to verify that the secure channel was established via the selected mechanism and that the EtherType of the first data frame sent between the TOE and the peer is 88-E5. Test 2: The evaluator shall capture traffic between the TOE and a MACsec-capable peer in the Operational Environment. The evaluator shall then cause the TOE to distribute a CAK to that peer, capture the MKPDUs from that operation, and verify the key is wrapped in the captured MKPDUs. Test 3: The evaluator shall set up an environment where the TOE is capable of communicating with two MACsec-capable peers in its Operational environment. The evaluator shall load a CAK into the TOE and the two peer devices, specifying a short lifetime, say, 10 minutes, and restore. The evaluator shall test two cases, one where the TOE is designated as the Key Server and principal actor, and one where it is the first peer (and also not a Key Server). The evaluator shall disconnect the second peer device, wait 10 minutes, and then reconnect the second peer. The evaluator shall verify in both cases that after 10 minutes, the Key Server will rekey the CA with the first peer, and then when the second peer is reconnected, the Key Server generates a new CAK that is distributed to that peer.

4.2.2.5 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.

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).

Application Note: The ICV has length 128 bits and is computed according to Section 9.4.1 of 802.1X. The ICV protects the destination and source MAC address parameters, as well as all the fields of the MAC Service Data Unit (MSDU) of the MKPDU including the allocated Ethertype, and up to but not including, the generated ICV.

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.

Assurance Activity	
TSS	The evaluator shall examine the TSS to verify that it describes the methods that the TOE implements to provide assurance of MKA integrity, including the use of an ICV and the ability to use a KDF to derive an ICK.
Test	The evaluator shall perform the following tests: Test 1: The evaluator shall transmit MKA traffic (MKPDUs) to the TOE from a MKA-capable

	peer in the Operational Environment. The evaluator shall verify via packet captures and/or audit logs that the last 16 octets of the MKPDUs in the received traffic do not appear to be predictable. Test 2: The evaluator shall transmit valid MKA traffic to the TOE from a MKA-capable peer in the operational environment that is routed through a test system set up as a man-in-the-middle. The evaluator shall use the test system to intercept this traffic to modify one bit in a packet payload before retransmitting to the TOE. The evaluator shall verify that the traffic is discarded due to an integrity failure.

FCS_ MKA_EXT.1.5 The TSF shall enforce an MKA Lifetime Timeout limit of 6.0 seconds and Hello Timeout limit of 2.0 seconds.

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

Application Note: The TSF functions correctly whether it acts as the Key Server or a peer device.

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.

Assurance Activity	
TSS	The evaluator shall verify that the TSS describes the TOE’s compliance with IEEE 802.1X-2010 and 802.1Xbx-2014 for MKA, including the values for MKA and Hello timeout limits and support for data delay protection. The evaluator shall also verify that the TSS describes the

	ability of the PAE of the TOE to establish unique CAs with individual peers and group CAs using a group CAK such that a new group SAK is distributed every time the group's membership changes. The evaluator shall also verify that the TSS describes the invalid MKPDUs that are discarded automatically by the TSF in a manner that is consistent with the SFR, and that valid MKPDUs are decoded in a manner consistent with IEEE 802.1X-2010 section 11.11.4.
AGD	The evaluator shall verify that the guidance documentation provides instructions on how to configure the TOE to act as the Key Server in an environment with multiple MACsec-capable devices.
Test	The tests below require the TOE to be deployed in an environment with two MACsec-capable peers, identified as devices B and C, that the TOE can communicate with. Prior to performing these tests, the evaluator shall follow the steps in the guidance documentation to configure the TOE as the Key Server and principal actor. The evaluator shall then perform the following tests: Test 1: The evaluator shall use a peer device to send MKA Hello traffic to the TOE, arbitrarily inducing artificial delays in their transmission using a man-in-the-middle setup. The evaluator shall observe that traffic delayed longer than 2.0 seconds is rejected. The evaluator shall repeat this test for MKA Lifetime traffic and observe that the timeout of 6.0 seconds is enforced by the TSF. Test 2: The evaluator shall establish an MKA session between the TOE and a peer device with a traffic sniffer set up to capture this traffic. The evaluator shall verify from packet captures that MKPDUs are sent at least once every half-second. Test 3: The evaluator shall perform the following steps: 1. Load one PSK onto the TOE and device B and a second PSK onto the TOE and device C. This defines two pairwise CAs. 2. Generate a group CAK for the group of 3 devices using ieee8021XKayCreateNewGroup. 3. Observe via packet capture that the TOE distributes the group CAK to the two peers, protected by AES key wrap using their respective PSKs. 4. Verify that B can form a SA with C and connect securely. 5. Disable the KaY functionality of device C using ieee8021XPaePortKayMkaEnable. 6. Generate a group CAK for the TOE and B using ieee8021XKayCreateNewGroup and observe they can connect. 7. The evaluator shall have B attempt to connect to C and observe this fails. 8. Re-enable the KaY functionality of device C. 9. Invoke ieee8021XKayCreateNewGroup again. 10. Verify that both the TOE can connect to C and that B can connect to C. Test 4: The evaluator shall start an MKA session between the TOE and the two environmental MACsec peers and then perform the following steps: 1. Send an MKPDU from the TOE to the individual MAC address of one peer. Verify the frame is dropped and logged. 2. Send an MKPDU from the TOE that is less than 32 octets long. Verify the frame is dropped and logged. 3. Send an MKPDU from the TOE whose length in octets is not a multiple of 4. Verify the frame is dropped and logged. 4. Send an MKPDU from the TOE that is one byte short. Verify the frame is dropped and

	logged. 5. Send an MKPDU from the TOE with unknown Agility Parameter. Verify the frame is dropped and logged.

4.2.2.6 FIA_AFL.1 Authentication Failure Handling

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

Application Note: The administrator should be able to set a time threshold for successive unsuccessful authentication attempts. After ten minutes has pass then the count of attempts can be reset to zero.

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been [met], the TSF shall [selection, choose one of: prevent the offending remote administrator from successfully authenticating until [*assignment: action*] is taken by a local Administrator; prevent the offending remote administrator from successfully authenticating until an Administrator defined time period has elapsed].

Application Note: This requirement does not apply to an administrator at a local console, since it does not make sense to lock a local administrator's account in this fashion. This could be addressed by (for example) requiring a separate account for local administrators or having the authentication mechanism implementation distinguish local and remote login attempts. The "action" taken by a local administrator is implementation specific and would be defined in the administrator guidance (for example, lockout reset or password reset). The ST author chooses one of the selections for handling of authentication failures depending on how the TOE has implemented this handler.

Assurance Activity	
TSS	The evaluator shall examine the TSS to determine that it contains a description, for each supported method for remote administrative actions, of how successive unsuccessful authentication attempts are detected and tracked. The TSS shall also describe the method by which the remote administrator is prevented from successfully logging on to the TOE, and the actions necessary to restore this ability.
AGD	The evaluator shall also examine the operational guidance to ensure that instructions for configuring the authentication failure threshold and the TOE's response to the threshold being met (if configurable), and that the process of allowing the remote administrator to once again successfully log on is described for each "action" specified (if that option is chosen). If different actions or mechanisms are implemented depending on the trusted path used to access the TSF (see FTP_TRP.1), all must be described.
Test	The evaluator shall perform the following tests for each method by which remote administrators access the TOE: Test 1: The evaluator shall use the operational guidance to configure the number of successive unsuccessful authentication attempts allowed by the TOE. The evaluator shall test that once the limit is reached for a given remote administrator account, subsequent attempts with valid credentials are not successful. Test 2: [conditional] If the TSS indicates that administrative action is necessary to re-enable an

	account that was locked out due to excessive authentication failures, the evaluator shall perform the steps in Test 1 to lock out an account, follow the operational guidance to manually re-enable the locked out administrator account, and observe that it is once again able to successfully log in. Test 3: [conditional] If the TSS indicates that an administrator-configurable time period must elapse in order to automatically re-enable an account that was locked out due to excessive authentication failures, the evaluator shall perform the steps in Test 1 to lock out an account, follow the operational guidance to configure a time period of their choosing, and observe through periodic login attempts that the account cannot successfully log in until the configured amount of time has elapsed. The evaluator shall then repeat this test for a different time period of their choosing.

4.2.2.7 FIA_PSK_EXT.1 Extended: Pre-Shared Key Composition

FIA_PSK_EXT.1.1 The TSF shall use pre-shared keys for MKA as defined by IEEE 802.1X, [selection: no other protocols, [*assignment: other protocols that use pre-shared keys*]].

FIA_PSK_EXT.1.2 The TSF shall be able to [selection: accept, generate using the random bit generator specified in FCS_RBG_EXT.1] bit-based pre-shared keys.

Application Note: For FIA_PSK_EXT.1.1, if other protocols can use pre-shared keys, they should be listed in the assignment as well; otherwise “no other protocols” should be chosen.

For FIA_PSK_EXT.12, the ST author specifies whether the TSF merely accepts bit-based pre-shared keys or if it is also capable of generating them. If it generates them, the requirement specifies that they must be generated using the RBG provided by the TOE.

Assurance Activity	
TSS	The evaluator shall examine the TSS to ensure that it identifies all protocols that allow both hex-based and bit-based pre-shared keys and that this conditioning is consistent with the last selection in the FIA_PSK_EXT.1.3 requirement. The evaluator shall also examine the TSS to ensure it describes the process by which the bit-based pre-shared keys are generated (if the TOE supports this functionality), and confirm that this process uses the RBG specified in FCS_RBG_EXT.1.
AGD	The evaluator shall examine the operational guidance to determine that it provides guidance to administrators on the composition of strong pre-shared keys, and (if the selection indicates keys of various lengths can be entered) that it provides information on the range of lengths supported The evaluator shall confirm the operational guidance contains instructions for either entering bit-based pre-shared keys for each protocol identified in the requirement, or generating a bit-

	based pre-shared key (or both). The evaluator shall also examine the TSS to ensure it describes the process by which the bit-based pre-shared keys are generated (if the TOE supports this functionality), and confirm that this process uses the RBG specified in FCS_RBG_EXT.1.
Test	The evaluator shall also perform the following tests for each protocol (or instantiation of a protocol, if performed by a different implementation on the TOE). Note that one or more of these tests can be performed with a single test case. Test 1 [conditional]: If the TOE supports pre-shared keys of multiple lengths, the evaluator shall use the minimum length; the maximum length; a length inside the allowable range; and invalid lengths beyond the supported range (both higher and lower). The minimum, maximum, and included length tests should be successful, and the invalid lengths must be rejected by the TOE. Test 2 [conditional]: If the TOE does not generate bit-based pre-shared keys, the evaluator shall obtain a bit-based pre-shared key of the appropriate length and enter it according to the instructions in the operational guidance. The evaluator shall then demonstrate that a successful protocol negotiation can be performed with the key. Test 3 [conditional]: If the TOE does generate bit-based pre-shared keys, the evaluator shall generate a bit-based pre-shared key of the appropriate length and use it according to the instructions in the operational guidance. The evaluator shall then demonstrate that a successful protocol negotiation can be performed with the key.

4.2.2.8 FPT_CAK_EXT.1 Protection of CAK Data

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

Application Note: The intent is for the TOE to protect CAK data from unauthorized disclosure. This data should only be accessed for the purposes of its assigned security functionality and there is no need for it to be displayed or accessed at any other time. This requirement does not prevent the device from providing indication that these exist, are in use, or are still valid. It does, however, restrict the reading of the values outright.

Assurance Activity	
TSS	The evaluator shall examine the TSS to determine that it details how CAKs are stored and that they are unable to be viewed through an interface designed specifically for that purpose. If these values are not stored in plaintext, the TSS shall describe how they are protected or obscured.
AGD	There are no guidance activities for this requirement.
Test	There are no test activities for this requirement.

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

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

Application Note: The intent of this requirement is to express the fail secure capabilities that the TOE possesses. This means that the TOE must be able to attain a secure/safe state (shutdown) when any of the identified failures occur.

Assurance Activity	
TSS	The evaluator shall examine the TSS to determine that it indicates that the TSF will shut down in the event that a self-test failure is detected.
AGD	The evaluator shall examine the operational guidance to verify that it describes the behavior of the TOE following a self-test failure and actions that an administrator should take if it occurs.
Test	The following test may require the vendor to provide access to a test platform that provides the evaluator with the ability to modify the TOE internals in a manner that is not provided to end customers: Test 1: The evaluator shall modify the TSF in a way that will cause a self-test failure to occur. The evaluator shall determine that the TSF shuts down and that the behavior of the TOE is consistent with the operational guidance. The evaluator shall repeat this test for each type of self-test that can be deliberately induced to fail.

4.2.2.10 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.

FPT_RPL.1.3 The TSF shall support extended packet numbering (XPN) as per IEEE 802.1AEbw-2013bw.

Application Note: As per IEEE 802.1AE-2006, replay is detected by examining the Packet Number (PN) value that is embedded in the Security TAG(SecTag) that is at the header of the MPDU. The PN is encoded in octets 5 through 8 of the SecTag to support replay protection. With XPN the PN is the least significant bits. The 32 most significant bits of the PN are recovered for each received frame by applying the assumption that they have remained unchanged since their use in the frame with the lowest acceptable PN unless the most significant of the 32 least significant bits of the lowest acceptable PN is set and the corresponding bit of the received PN is not set, in which case the value of the 32 most significant bits of the PN is one more than the value of the 32 most significant bits of the lowest acceptable PN.

Assurance Activity	
TSS	The evaluator shall examine the TSS to determine that it describes how replay is detected for MPDUs and how replayed MPDUs are handled by the TSF.
AGD	There are no guidance activities for this requirement.
Test	The evaluator shall perform the following tests:

	Test 1: The evaluator shall set up a MACsec connection with an entity in the Operational Environment. The evaluator shall then capture traffic sent from this remote entity to the TOE. The evaluator shall retransmit copies of this traffic to the TOE in order to impersonate the remote entity where the PN values in the SecTag of these packets are less than the lowest acceptable PN for the SA. The evaluator shall observe that the TSF does not take action in response to receiving these packets and that the audit log indicates that the replayed traffic was discarded. Test 2: The evaluator will capture frames during a MKA session and record the lowest PN observed in a particular time range. The evaluator will then send a frame with a lower PN, then verify that this frame is dropped. The evaluator will verify that the device logged this event. Test 3: The evaluator shall configure the TOE to use the GCM-AES-XPN-128 cipher suite and repeat Test 1 and Test 2. Test 4: The evaluator shall configure the TOE to use the GCM-AES-XPN-256 cipher suite and repeat Test 1 and Test 2.

Appendix A - Rationale

In this EP, the focus in the initial sections of the document is to use a narrative presentation in an attempt to increase the overall understandability of the threats addressed by MACsec devices; the methods used to mitigate those threats; and the extent of the mitigation achieved by compliant TOEs. This presentation style does not readily lend itself to a formalized evaluation activity, so this section contains the tabular artifacts that can be used for the evaluation activities associated with this document.

A.1 Security Problem Definition

A.1.1 Assumptions

No assumptions are defined for this EP. As an extended package to the NDcPP, the TOE inherits all assumptions defined by the base PP, with one exception as defined in Section 2.2 of this EP.

A.1.2 Threats

The threats listed below are addressed by MACsec devices. Note that these threats are in addition to those defined in the NDcPP, all of which apply to MACsec devices.

Threat Name	Threat Definition
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.

Table 2: Threats

A.1.3 Organizational Security Policies

No organizational policies have been identified that are specific to MACsec devices. However, all the organizational security policies in the NDcPP apply to MACsec devices.

A.1.4 Security Problem Definition Correspondence

The following table serves to map the threats and assumptions defined in this EP to the security objectives also defined or identified in this EP:

Threat or Assumption	Security Objectives
T.DATA_INTEGRITY	O.CRYPTOGRAPHIC_FUNCTIONS, O.REPLAY_DETECTION
T.NETWORK_ACCESS	O.PORT_FILTERING
T.UNTRUSTED_COMMUNICATION_CHANNELS	O.CRYPTOGRAPHIC_FUNCTIONS, O.AUTHENTICATION

Table 3: Security Problem Definition Correspondence

Note that this EP also includes security objectives that address threats from the base NDcPP in a more refined manner, based on the specific functions provided by a MACsec Ethernet Encryption TOE, as follows:

- O.SYSTEM_MONITORING further mitigates NDcPP threat T.UNDETECTED_ACTIVITY
- O.AUTHORIZED_ADMINISTRATION further mitigates NDcPP threats T.UNAUTHORIZED_ADMINISTRATOR_ACCESS and T.SECURITY_FUNCTIONALITY_COMPROMISE
- O.TSF_INTEGRITY further mitigates NDcPP threat T.SECURITY_FUNCTIONALITY_FAILURE
- O.VERIFIABLE UPDATES further mitigates NDcPP threat T.UPDATE_COMPROMISE

A.2 Security Objectives

A.2.1 Security Objectives for the TOE

The following table contains security objectives specific to MACsec devices:

Objective Name	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.

Table 4: Security Objectives for the TOE

A.2.2 Security Objectives for the Operational Environment

No environmental security objectives are defined for this EP. As an extended package to the NDcPP, the TOE inherits all environmental security objectives defined by the base PP, with one exception as defined in Section 3.2 of this EP.

A.2.3 Security Objective Correspondence

The correspondence between the Security Functional Requirements (SFRs) and Security Objectives identified or defined in this EP is provided in Section 3.

Appendix B – Optional Requirements

As indicated in the introduction to this EP, the baseline requirements are contained in the body of this EP. There are additional requirements that can be included in the ST, but do not have to be in order for a TOE to claim conformance to this EP.

B.1 FIA_AFL_EXT.1 Extended: Authentication Attempt Limiting

FIA_AFL_EXT.1.1 When 3 unsuccessful authentication attempts have been made to the local console, the TSF shall limit the rate of login attempts to one per minute.

Application Note: This requirement applies to an administrator at a local console. This anti-hammering requirement is to slow down brute force password guessing.

Assurance Activity	
TSS	The evaluator shall examine the TSS to determine that it describes the ability of the TSF to limit the rate at which authentication attempts can be made at the local console following three successive failed attempts.
AGD	If the TOE requires configuration to be put into a state where authentication attempt limiting is enforced, the evaluator shall review the operational guidance to verify that it describes the procedures to configure the TOE into this state.
Test	The evaluator shall follow the operational guidance to configure the TOE into a state that enforces authentication attempt limiting (if applicable). The evaluator shall successfully log in to the TOE at a local console, log back out, and immediately log back in in order to demonstrate that successive authentication attempts can be made in under a minute. The evaluator shall then enter an incorrect password three consecutive times for the same account to trigger authentication attempt limiting. Once the TOE is in this state, the evaluator shall attempt to log in to the TOE periodically over several attempts of varying time intervals and observe that authentication attempts cannot be made any more frequently than once per minute.

Appendix C – Selection-Based Requirements

As indicated in the introduction to this EP, the baseline requirements (those that must be performed by the TOE or its underlying platform) are contained in the body of this EP. There are additional requirements based on selections in the body of the EP; if certain selections are made, then additional requirements below will need to be included.

C.1 FMT_SNMP_EXT.1 SNMP Management

The following SFRs shall be included by the ST author if SNMPv3 is selected to provide a trusted channel to a remote administrator in FTP_TRP.1.1.

FMT_SNMP_EXT.1.1 The TSF shall implement Simple Network Management Protocol (SNMP) with TLS security in conformance with RFC 6353 "Transport Layer Security (TLS) Transport Model for the Simple Network Management Protocol (SNMP)" and to an authorized IT entity in FTP_ITC.1.

FMT_SNMP_EXT.1.2 The TSF shall permit access to TSF management functions using only SNMP version 3.

FMT_SNMP_EXT.1.3 The TSF shall support the following password quality metrics for SNMPv3 passwords: [*character selections and minimum length defined in FIA_PMG_EXT.1*].

Application Note: FIA_PMG_EXT.1 is defined in the NDcPP so there will not be a MACsec TOE that does not claim this SFR.

Assurance Activity	
TSS	The evaluator shall examine the TSS to determine that it describes the ability of the TSF to support SNMPv3 for remote management for connections to authorized IT entities (per FTP_ITC.1)and that it can apply appropriate password restrictions to this interface.
AGD	If the TOE requires configuration to be put into a state where SNMPv3 is the only version of SNMP that is accepted, the evaluator shall verify that the operational guidance provides instructions on how to disable unsupported versions of SNMP.
Test	The evaluator shall configure the TOE in accordance with its operational guidance to accept no versions of SNMP other than SNMPv3 (if applicable). The evaluator shall then perform the following tests: Test 1: The evaluator shall attempt to connect to the TOE using SNMPv2 and observe that the connection is not successful. Test 2: The evaluator shall attempt to connect to the TOE using SNMPv1 and observe that the connection is not successful. Test 3: The evaluator shall attempt to set a password for the SNMP management interface

	that contains characters that are not listed in FIA_PMG_EXT.1 in the NDcPP and observe that the password cannot be set to this value. The evaluator shall repeat this test for a password that does not meet minimum length requirements as described by FIA_PMG_EXT.1 in the NDcPP and observe that the password cannot be set to this value. Testing of the security of the SNMPv3 trusted path is tested as part of FTP_TRP.1 and testing of the ability to manage the TSF using SNMPv3 is tested as part of FMT_SMF.1.

FCS_SNMP_EXT.1.1 The evaluator will verify that the implementation of SNMP is configured to operate with confidentiality and integrity protection using TLS with cipher suites that support AES (128 and 256 bit key sizes) and SHA algorithms.

C.2 FCS_EAP-TLS_EXT.1 EAP-TLS Protocol

The following SFR shall be included by the ST author if EAP-TLS with DevIDs is selected as an authentication method in FCS_MACSEC_EXT.4.1:

FCS_EAP-TLS_EXT.1.1 The TSF shall implement the Extensible Authentication Protocol (EAP) (RFC 3748) and EAP-Transport Layer Security (EAP-TLS) (RFC 5216).

FCS_EAP-TLS_EXT.1.2 The TSF shall implement TLS 1.1 (RFC4346 and [selection: TLS 1.2 (RFC 5246), no other TLS version] supporting the following ciphersuites: [

- Mandatory Ciphersuites:
  - TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 5246
- [selection: Optional Ciphersuites:
  - TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 5246
  - TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 5246
  - TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 5246
  - TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492
  - TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492
  - TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492
  - TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492
  - TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246
  - TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246
  - TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256 as defined in RFC 5246
  - TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246
  - TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289
  - TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289
  - TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
  - TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289
  - no other ciphersuite]].

Assurance Activity	
TSS	The evaluator shall check the TSS to verify that it describes the ability of the TSF to support EAP-TLS along with the TLS versions and ciphersuites supported in the EAP-TLS implementation.
AGD	If the TLS version and/or supported ciphersuites are configurable, the evaluator shall review

	the operational guidance to verify that it provides instructions on how these are configured.
Test	The evaluator shall set up an environment where the TOE can connect to a second MACsec device, identified as device B. The evaluator shall configure the devices in two cases: first where the TOE will be the Authenticator and device B will be the Supplicant, and second where the TOE will be the Supplicant and device B will be the Authenticator. The evaluator shall set up an Authentication Server, which may run on the TOE or be a separate device that connects to the test environment. The evaluator shall then perform the following tests: 1. The evaluator will cause the Supplicant to initiate an EAP-TLS session with the Authenticator. 2. The evaluator will intercept, manipulate, and retransmit the first packet sent by the Supplicant. 3. The evaluator will increment the length field and verify that the Authenticator does not respond (i.e. silently discards the packet). The evaluator will append at least one octet to the end of the packet and verify that the Authenticator responds as if there were no change (i.e., ignores the additional octets).

C.3 FCS_DEVID_EXT.1 Secure Device Identifiers

The following SFR shall be included by the ST author if EAP-TLS with DevIDs is selected as an authentication method in FCS_MACSEC_EXT.4.1:

FCS_DEVID_EXT.1.1 The TSF shall implement Secure Device Identifiers (DevIDs) following IEEE Standard 802.1AR-2009.

FCS_DEVID_EXT.1.2 The TSF shall contain an Initial DevID (IDevID) as specified in Section 6.2.1 of IEEE 802.1AR-2009.

FCS_DEVID_EXT.1.3 The TSF shall contain the credential chain as specified in Section 6.2.1 of IEEE 802.1AR-2009.

FCS_DEVID_EXT.1.4 The TSF shall verify that both the Supplicant and Authenticator DevIDs presented for EAP-TLS have credentials that chain to one of the specified Certificate Authorities.

FCS_DEVID_EXT.1.5 The TSF shall not establish a trusted channel if the Supplicant DevID is invalid.

Assurance Activity	
TSS	The evaluator shall check the TSS to verify that it describes how the TSF implements and validates DevIDs.
AGD	There are no operational guidance activities for this requirement.
Test	The evaluator shall perform the following tests: Test 1: 1. The evaluator shall install a DevID in the Supplicant that has one octet changed to invalidate the signature.

	2. The evaluator will cause the Supplicant to initiate an EAP-TLS session with the Authenticator. 3. The evaluator will verify that the connection fails. Test 2: 1. The evaluator shall install a DevID in the Supplicant with a valid signature but from an issuer not recognized by the Authenticator. 2. The evaluator will cause the Supplicant to initiate an EAP-TLS session with the Authenticator. 3. The evaluator will verify that the connection fails. Test 3: 1. The evaluator will cause the Supplicant to initiate an EAP-TLS session with the Authenticator. 2. The evaluator will intercept, manipulate, and retransmit the packets sent by the Supplicant so that the presented name differs from the name in the DevID. 3. The evaluator will verify that the connection fails.

FCS_DEVID_EXT.1.6 The TSF shall support mutual authentication using DevIDs.

Assurance Activity	
TSS	The evaluator shall check the TSS to verify that describes the ability of the TSF to support mutual authentication using DevIDs.
AGD	There are no operational guidance activities for this requirement.
Test	The evaluator shall perform the following tests: Test 1: 1. The evaluator will cause the Supplicant to initiate an EAP-TLS session with the Authenticator in which mutual authentication is requested. 2. The evaluator will verify that the EAP-TLS packet with a Client Certificate Request message is sent and that the Supplicant responds with its DevID.

FCS_DEVID_EXT.1.7 The TSF shall support the following operations as specified in Section 6.3 of IEEE 802.1AR-2009:

1. Enable/disable DevID credential
2. Enable/disable DevID key

Assurance Activity	
TSS	The evaluator shall check the TSS to verify that describes the ability of the TSF to support the signing, enable/disable DevID credential, and enable/disable DevID key operations.
AGD	There are no operational guidance activities for this requirement.
Test	The evaluator shall perform the following tests: Test 1: 1. The evaluator will disable the Supplicant public key by setting MIB object devIDPublicKeyEnabled to false. 2. The evaluator will cause Supplicant to initiate an EAP-TLS session with the

	Authenticator. 3. The evaluator will verify that the Supplicant is unable to authenticate. 4. The evaluator will re-enable the public key, then verify the Supplicant can authenticate. Test 2: 1. The evaluator will disable the Supplicant DevID by setting MIB object devIDCredentialEnabled to false. 2. The evaluator will cause Supplicant to initiate an EAP-TLS session with the Authenticator. 3. The evaluator will verify that the Supplicant is unable to authenticate. 4. The evaluator will re-enable the DevID, then verify the Supplicant can authenticate.

Appendix D – Objective Requirements

As indicated in the introduction to this EP, the baseline requirements (those that must be performed by the TOE or its underlying platform) are contained in the body of this EP. There are additional requirements that specify security functionality that is desirable and these requirements are contained in this Appendix. It is expected that these requirements will transition from objective requirements to baseline requirements in future versions of this EP.

At this time no objective requirements specific to MACsec TOEs have been identified.