Persistent Systems LLC, Wave Relay® Devices v1.0 Security Target

Version 1.0
March 24, 2025

Prepared for:

PERSISTENT SYSTEMS

www.persistentsystems.com

601 West 26th St, Suite 905
New York, NY 10006

Prepared By:

Gossamer Laboratories

www.gossamersec.com

LIST OF TABLES

1 Security Target Introduction

This section identifies the Security Target (ST) and Target of Evaluation (TOE) identification, ST conventions, ST conformance claims, and the ST organization. The TOE is Wave Relay provided by Persistent Systems. The TOE is being evaluated as a network device (and IPsec VPN Gateway and MACsec Ethernet Encryption device).

The Security Target contains the following additional sections:

- Conformance Claims (Section 2)
- Security Objectives (Section 3)
- Extended Components Definition (Section 4)
- Security Requirements (Section 5)
- TOE Summary Specification (Section 6)

Conventions

The following conventions have been applied in this document:

- Security Functional Requirements – Part 2 of the CC defines the approved set of operations that may be applied to functional requirements: iteration, assignment, selection, and refinement.
    - Iteration: allows a component to be used more than once with varying operations. In the ST, iteration is indicated by a parenthetical number placed at the end of the component. For example FDP_ACC.1(1) and FDP_ACC.1(2) indicate that the ST includes two iterations of the FDP_ACC.1 requirement.
    - Assignment: allows the specification of an identified parameter. Assignments are indicated using bold and are surrounded by brackets (e.g., [**assignment**]). Note that an assignment within a selection would be identified in italics and with embedded bold brackets (e.g., [***[selected-assignment]***]).
    - Selection: allows the specification of one or more elements from a list. Selections are indicated using bold italics and are surrounded by brackets (e.g., [***selection***]).
    - Refinement: allows the addition of details. Refinements are indicated using bold, for additions, and strike-through, for deletions (e.g., "… **all** objects …" or "… ~~some~~ **big** things …").
- Other sections of the ST – Other sections of the ST use bolding to highlight text of special interest, such as captions.

1.1 Security Target Reference

ST Title – Persistent Systems LLC, Wave Relay® Devices v1.0 Security Target

ST Version – Version 1.0

ST Date – March 24, 2025

1.2 TOE Reference

TOE Identification – Persistent Systems LLC, Wave Relay® Devices v1.0 running Wave Relay OS 2.2

TOE Developer – Persistent Systems, LLC

Evaluation Sponsor – Persistent Systems, LLC

1.3 TOE Overview

The Target of Evaluation (TOE) is Persistent Systems LLC, Wave Relay® Devices v1.0 running Wave Relay OS 2.2. The TOE provides secure, seamless ethernet connectivity, ensuring global connectivity for users in any location and

under any circumstances. By establishing a resilient, secure connectivity fabric, the TOE enables mission-critical communication, regardless of geographical constraints or operational challenges.

1.4 TOE Description

The TOE leverages a custom OS called Wave Relay OS that provides a secure operating environment. Available as a hardware network appliance, the TOE supports a wide range of network, wireless and security protocols designed for peer-to-peer MANET networking at OSI Layer 2 and Layer 3. This includes, for instance, the use of multiple layer-3 Gateways in a MANET.

The TOE is capable of securing communication via its ethernet interface with MACsec, IPsec and TLS. Remote administration utilizes TLS to protect communications to the Wave Relay Device GUI and programmatic interface.

For the purposes of evaluation, the TOE will be treated as a Network Device, IPsec VPN Gateway and MACsec Ethernet Encryption Device. Thus, the security functionality offered by the TOE includes validated secure by design components such as CAVP tested Cryptographic support, Trusted updates, Self Tests, Secure connections, Identification & Authentication, Packet Filtering, and Secure Auditing.

It is important to note that functions outside the scope of NDcPP22e/MACSEC10/VPNGW13 were not evaluated.

1.4.1 TOE Architecture

The TOE is a hardware network appliance available in several models with varying form factors (See Table 1-1 TOE Models).

Model	Processor
MPU5 (WR-5100)	NXP iMX6
Embedded Module (WR-5200)	NXP i.MX6
Embedded Module Lite (WR-5250)	NXP i.MX6
GVR5 (WR-GVR5-SYS)	NXP i.MX6
Integrated Antenna Series (WR-INT-ANT-SYS)	NXP i.MX6

Table 1-1 TOE Models

The Man Portable Unit Generation 5 (MPU5) is a wearable Wave Relay device that can be connected to a wired network and securely communicate with network infrastructure services such as external audit servers, management stations, as well as VPN peers.

The Embedded Module and Embedded Module Lite are Wave Relay embedded devices in a SWaP-timized form factor designed to transform your UAS, UGV platform into a networked asset.

The GVR5 model is a dual band Wave Relay solution, engineered to for tracked and wheeled ground vehicles as well as aircraft.

The Integrated Antenna Series applies the power of Wave Relay directly into an antenna extending the enterprise to the edge of large geographic areas.

The multiple TOE appliance models are designed to support different mission requirements while using the same ARM Cortex-A9 Architecture. NXP i.MX6 series is a family of ARM-based processors designed for a variety of applications balancing power efficiency, performance and flexibility.

While Persistent Systems Wave Relay products can be configured as a collection of independent devices operating in a network, the TOE configuration subject to this evaluation is limited to a single Wave Relay device.

A Persistent Systems Wave Relay device is a network appliance with NXP iMX6 Cortex-A9 CPU running software designed to provide the required capabilities. All Wave Relay Devices include the same validated cryptographic providers that are used to perform cryptographic functions for TLS, IPsec, and MACsec.

- Wave Relay® Kernel Space Crypto Module (HW) version 1.0 (Cert. #A4588)

- Wave Relay® Kernel Space Crypto Module (SW) version 1.0 (Cert. #A4589) All algorithms are supported with PAA and without PAA.
- Wave Relay® User Space Crypto Module version 1.0 (Cert. #A5177). All algorithms are supported with PAA and without PAA

1.4.1.1 Physical Boundaries

The physical boundaries of the TOE consist of the physical boundaries of the Persistent Systems Wave Relay device.

1.4.1.2 Logical Boundaries

This section summarizes the security functions provided by Persistent Systems Wave Relay Devices:

- Security audit
- Cryptographic support
- Identification and authentication
- Security management
- Packet Filtering
- Protection of the TSF
- TOE access
- Trusted path/channels

1.4.1.2.1 Security audit

The TOE generates audit events for numerous activities including events related to cryptographic functionality, identification and authentication, and administrative actions. The TOE generates a complete audit record including the IP address of the TOE, the event details, and the time the event occurred. The TOE provides the administrator with a local circular audit trail where the TOE overwrites the oldest audit records with the newest audit records when space is full. Audit logs are also sent to a remote syslog server in the environment over TLS encrypted channel.

1.4.1.2.2 Cryptographic support

The TOE provides cryptography in support of other TOE security functionality. The TOE provides cryptography in support of secure connections using IPsec, TLS, MACsec and remote administrative management HTTPS/TLS.

1.4.1.2.3 Identification and authentication

The TOE allows unauthenticated users to read the login banner, view the TOE identity (DNS name and IP address), view the TOE power level, and view status. The TOE also performs packet filtering operations prior to administrator login. The TOE requires users to be authenticated before all other administrative operations.

The TOE authenticates the administrator prior to granting access to the GUI and programmatic interfaces by accepting a password. The TOE supports the validation of x509v3 certificates for authentication in the context of the TLS and IPsec protocols. These certificates can be ECDSA certificates. The TOE also supports pre-shared key authentication for MACsec and IPsec connections. The TOE checks the revocation status of a certificate using OCSP or CRLs.

1.4.1.2.4 Security management

Security management commands are limited to authorized users (i.e., administrators) and available only after they have provided acceptable user identification and authentication data to the TOE. All TOE administration occurs through a TLS/HTTPS session.

1.4.1.2.5 Packet Filtering

The TOE provides packet filtering and secure IPsec tunneling functionality. The tunnels can be established between the TOE and a VPN peer. An authorized administrator can define the traffic that needs to be protected via IPsec by configuring access lists (permit, deny, log) and applying these access lists to the VPN interfaces.

1.4.1.2.6 Protection of the TSF

The TOE provides a variety of means of protecting itself. The TOE performs self-tests and integrity verification that cover the correct operation of the TOE at startup. Any test failures that occur will prevent the TOE from booting to a usable state. It provides functions necessary to securely update the TOE. The TOE includes a hardware clock to ensure reliable timestamps. The TOE's time can be configured manually or by syncing to a remote NTP server. It protects sensitive data such as stored passwords and cryptographic keys so that they are not accessible through the TOE, even to a Security Administrator.

The TOE has the ability detect replay of frames received over the MACsec channel. The detected replayed frames are dropped.

1.4.1.2.7 TOE access

The TOE can be configured to display a logon banner before a user session is established. The TOE also enforces inactivity timeouts for local and remote sessions that can be configured by an administrator.

1.4.1.2.8 Trusted path/channels

The TOE protects interactive communication with administrators using TLS for GUI and programmatic access. The TLS protocol provides integrity and disclosure protection. If the negotiation of a TLS session fails, the attempted connection will not be established.

The TOE protects communication with network peers, such as an external audit server (syslog server) and a VPN peer using IPsec connections to provide disclosure or modification protections. The TOE can be configured to use MACsec to secure the channel to an external audit server (syslog server) at Layer 2. The TOE can also provide a TLS connection to a controlled network device and validate the X509v3 certificate that is presented by the device.

1.4.2 TOE Documentation

The following administrator and user guidance are available:

Common Criteria Administrator Guide, Target of Evaluation: Persistent Systems LLC, Wave Relay® Devices v1.0, version 1.0, March 24, 2025

2 Conformance Claims

This TOE is conformant to the following CC specifications:

- Common Criteria for Information Technology Security Evaluation Part 2: Security functional components, Version 3.1, Revision 5, April 2017.
    - Part 2 Extended
- Common Criteria for Information Technology Security Evaluation Part 3: Security assurance components, Version 3.1, Revision 5, April 2017.
    - Part 3 Conformant
- Package Claims:
    - PP-Configuration for Network Devices, MACsec Ethernet Encryption, and VPN Gateways, Version 1.1, August 18, 2023
        - Base-PP: collaborative Protection Profile for Network Devices, Version 2.2e (CPP_ND_V2.2E)
        - PP-Module: PP-Module for MACsec Ethernet Encryption, Version 1.0 (MOD_MACsec_V1.0)
        - PP-Module: PP-Module for VPN Gateways, Version 1.3 (MOD_VPNGW_V1.3)
- For shorthand, this ST will use the following abbreviations for the PPs and Modules:
    - CPP_ND_V2.2E – NDcPP22e
    - MOD_MACsec_V1.0 – MACSEC10
    - MOD_VPNGW_V1.3 – VPNGW13
    - (NDcPP22e/MACSEC10/VPNGW13)
- Technical Decisions:

Package	Technical Decision	Applied	Notes
NDcPP22e	TD0800: Updated NIT Technical Decision for IPsec IKE/SA Lifetimes Tolerance	Yes	
NDcPP22e	TD0792: NIT Technical Decision: FIA_PMG_EXT.1: TSS EA not in line with SFR	Yes	
NDcPP22e	TD0790: NIT Technical Decision: Clarification Required for testing IPv6	Yes	
NDcPP22e	TD0738: NIT Technical Decision for Link to Allowed-With List	Yes	
NDcPP22e	TD0670: NIT Technical Decision for Mutual and Non-Mutual Auth TLSC Testing	Yes	
NDcPP22e	TD0639: NIT Technical Decision for Clarification for NTP MAC Keys	Yes	
NDcPP22e	TD0638: NIT Technical Decision for Key Pair Generation for Authentication	Yes	
NDcPP22e	TD0636: NIT Technical Decision for Clarification of Public Key User Authentication for SSH	No	FCS_SSHC_EXT.1 not claimed
NDcPP22e	TD0635: NIT Technical Decision for TLS Server and Key Agreement Parameters	Yes	
NDcPP22e	TD0632: NIT Technical Decision for Consistency with Time Data for vNDs	Yes	
NDcPP22e	TD0631: NIT Technical Decision for Clarification of public key authentication for SSH Server	Yes	
NDcPP22e	TD0592: NIT Technical Decision for Local Storage of Audit Records	Yes	
NDcPP22e	TD0591: NIT Technical Decision for Virtual TOEs and hypervisors	Yes	

Package	Technical Decision	Applied	Notes
NDcPP22e	TD0581: NIT Technical Decision for Elliptic curve-based key establishment and NIST SP 800-56Arev3	Yes	
NDcPP22e	TD0580: NIT Technical Decision for clarification about use of DH14 in NDcPPv2.2e	Yes	
NDcPP22e	TD0572: NiT Technical Decision for Restricting FTP_ITC.1 to only IP address identifiers	Yes	
NDcPP22e	TD0571: NiT Technical Decision for Guidance on how to handle FIA_AFL.1	Yes	
NDcPP22e	TD0570: NiT Technical Decision for Clarification about FIA_AFL.1	Yes	
NDcPP22e	TD0569: NIT Technical Decision for Session ID Usage Conflict in FCS_DTLSS_EXT.1.7	Yes	
NDcPP22e	TD0564: NiT Technical Decision for Vulnerability Analysis Search Criteria	Yes	
NDcPP22e	TD0563: NiT Technical Decision for Clarification of audit date information	Yes	
NDcPP22e	TD0556: NIT Technical Decision for RFC 5077 question	Yes	
NDcPP22e	TD0555: NIT Technical Decision for RFC Reference incorrect in TLSS Test	Yes	
NDcPP22e	TD0547: NIT Technical Decision for Clarification on developer disclosure of AVA_VAN	Yes	
NDcPP22e	TD0546: NIT Technical Decision for DTLS: clarification of Application Note 63	No	FCS_DTLSC_EXT.1 is not claimed
NDcPP22e	TD0537: NIT Technical Decision for Incorrect reference to FCS_TLSC_EXT.2.3	No	FCS_TLSC_EXT.2 not claimed
NDcPP22e	TD0536: NIT Technical Decision for Update Verification Inconsistency	Yes	
NDcPP22e	TD0528: NIT Technical Decision for Missing EAs for FCS_NTP_EXT.1.4	No	
NDcPP22e	TD0527: Updates to Certificate Revocation Testing (FIA_X509_EXT.1)	Yes	
VPNGW13	TD0878: Updating FIPS 186-4 to 186-5 in MOD_VPNGW_V1.3	Yes	
VPNGW13	TD0838: PPK Configurability in FIA_PSK_EXT.1.1	Yes	
VPNGW13	TD0824: Aligning MOD_VPNGW 1.3 with NDcPP 3.0E	Yes	
VPNGW13	TD0811: Correction to Referenced SFR in FIA_PSK_EXT.3 Test	Yes	
VPNGW13	TD0781: Correction to FIA_PSK_EXT.3 EA for MOD_VPNGW_v1.3	Yes	
MACSEC10	TD0891: Correlation of Implicitly Satisfied Requirements when CPP_ND_V3.0E is the Base-PP	No	NDcPP30e not used
MACSEC10	TD0889: Correction For Tests Incorrectly Requiring Group MACsec	Yes	
MACSEC10	TD0884: Expansion of Permitted EtherTypes in FCS_MACSEC_EXT.1.4	Yes	
MACSEC10	TD0882: MACsec Data Delay Protection, Key Agreement, and Conditional Support for Group CAK	Yes	
MACSEC10	TD0881: Correction to MN Usage for FPT_RPL.1 Test	Yes	
MACSEC10	TD0870: Security Objectives Rationale for MOD_MACSEC_V1.0	Yes	
MACSEC10	TD0869: Correction to MN Usage for FPT_RPL.1 Test	Yes	
MACSEC10	TD0840: Alignment of Test 22.1 to FMT_SMF.1/MACSEC	Yes	
MACSEC10	TD0826: Aligning MOD_MACSEC_V1.0 with CPP_ND_V3.0E	Yes	
MACSEC10	TD0816 : Clarity for MACsec Self Test Failure Response	Yes	
MACSEC10	TD0803: Clarification for Configurable MACsec CKN Length	Yes	
MACSEC10	TD0746: Correction to FPT_RPL.1 Test 25	Yes	
MACSEC10	TD0728: Corrections to MACSec PP-Module SD	Yes	

2.1 Conformance Rationale

The ST conforms to the NDcPP22e/MACSEC10/VPNGW13. As explained previously, the security problem definition, security objectives, and security requirements have been drawn from the PP.

3 Security Objectives

The Security Problem Definition may be found in the NDcPP22e/MACSEC10/VPNGW13 and this section reproduces only the corresponding Security Objectives for operational environment for reader convenience. The NDcPP22e/MACSEC10/VPNGW13 offers additional information about the identified security objectives, but that has not been reproduced here and the NDcPP22e/MACSEC10/VPNGW13 should be consulted if there is interest in that material.

In general, the NDcPP22e/ MACSEC10/ VPNGW13 has defined Security Objectives appropriate for a network device providing secure communication channels and as such are applicable to the Wave Relay Device TOE.

3.1 Security Objectives for the Operational Environment

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.COMPONENTS_RUNNING (applies to distributed TOEs only)
For distributed TOEs, the Security Administrator ensures that the availability of every TOE component is checked as appropriate to reduce the risk of an undetected attack on (or failure of) one or more TOE components. The Security Administrator also ensures that it is checked as appropriate for every TOE component that the audit functionality is running properly.

OE.CONNECTIONS See TD0520 for SARs.
The TOE is connected to distinct networks in a manner that ensures that the TOE security policies will be enforced on all applicable network traffic flowing among the attached networks.

OE.NO_GENERAL_PURPOSE There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE. Note: For vNDs the TOE includes only the contents of the its own VM, and does not include other VMs or the VS.

OE.NO_THRU_TRAFFIC_PROTECTION The TOE does not provide any protection of traffic that traverses it. It is assumed that protection of this traffic will be covered by other security and assurance measures in the operational environment.

OE.PHYSICAL Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.

OE.RESIDUAL_INFORMATION The Security Administrator ensures that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment. For vNDs, this applies when the physical platform on which the VM runs is removed from its operational environment.

OE.TRUSTED_ADMIN TOE 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.VM_CONFIGURATION (applies to vNDs only)
For vNDs, the Security Administrator ensures that the VS and VMs are configured to

- reduce the attack surface of VMs as much as possible while supporting ND functionality (e.g., remove unnecessary virtual hardware, turn off unused inter-VM communications mechanisms), and
- correctly implement ND functionality (e.g., ensure virtual networking is properly configured to support network traffic, management channels, and audit reporting).

The VS should be operated in a manner that reduces the likelihood that vND operations are adversely affected by virtualization features such as cloning, save/restore, suspend/resume, and live migration.

If possible, the VS should be configured to make use of features that leverage the VS's privileged position to provide additional security functionality. Such features could include malware detection through VM introspection, measured VM boot, or VM snapshot for forensic analysis.

4 Extended Components Definition

All of the extended requirements in this ST have been drawn from the NDcPP22e/MACSEC10/VPNGW13. The NDcPP22e/MACSEC10/VPNGW13defines the following extended requirements and since they are not redefined in this ST the NDcPP22e/MACSEC10/VPNGW13 should be consulted for more information in regard to those CC extensions.

Extended SFRs:

- NDcPP22e:FAU_STG_EXT.1: Protected Audit Event Storage
- NDcPP22e:FCS_HTTPS_EXT.1: HTTPS Protocol
- NDcPP22e:FCS_IPSEC_EXT.1: IPsec Protocol
- VPNGW13:FCS_IPSEC_EXT.1: IPsec Protocol - per TD0657
- MACsecEP12:FCS_MACSEC_EXT.1: MACsec
- MACsecEP12:FCS_MACSEC_EXT.2: MACsec Integrity and Confidentiality
- MACsecEP12:FCS_MACSEC_EXT.3: MACsec Randomness
- MACsecEP12:FCS_MACSEC_EXT.4: MACsec Key Usage
- MACsecEP12:FCS_MKA_EXT.1: MACsec Key Agreement
- NDcPP22e:FCS_NTP_EXT.1: NTP Protocol
- NDcPP22e:FCS_RBG_EXT.1: Random Bit Generation
- NDcPP22e:FCS_TLSC_EXT.1: TLS Client Protocol Without Mutual Authentication - per TD0670
- NDcPP22e:FCS_TLSS_EXT.1: TLS Server Protocol Without Mutual Authentication - per TD0635
- NDcPP22e:FIA_PMG_EXT.1: Password Management
- MACsecEP12:FIA_PSK_EXT.1: Extended: Pre-Shared Key Composition
- VPNGW13:FIA_PSK_EXT.1: Pre-Shared Key Composition
- NDcPP22e:FIA_UAU_EXT.2: Password-based Authentication Mechanism
- NDcPP22e:FIA_UIA_EXT.1: User Identification and Authentication
- NDcPP22e:FIA_X509_EXT.1/Rev: X.509 Certificate Validation
- VPNGW13:FIA_X509_EXT.1/Rev: X.509 Certificate Validation
- NDcPP22e:FIA_X509_EXT.2: X.509 Certificate Authentication
- VPNGW13:FIA_X509_EXT.2: X.509 Certificate Authentication
- NDcPP22e:FIA_X509_EXT.3: X.509 Certificate Requests
- VPNGW13:FIA_X509_EXT.3: X.509 Certificate Requests
- VPNGW13:FPF_RUL_EXT.1: Packet Filtering Rules - per TD0683
- NDcPP22e:FPT_APW_EXT.1: Protection of Administrator Passwords
- MACsecEP12:FPT_CAK_EXT.1: Protection of CAK Data
- NDcPP22e:FPT_SKP_EXT.1: Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)
- NDcPP22e:FPT_STM_EXT.1: Reliable Time Stamps - per TD0632
- NDcPP22e:FPT_TST_EXT.1: TSF testing
- VPNGW13:FPT_TST_EXT.1: TSF Testing
- VPNGW13:FPT_TST_EXT.3: Self-Test with Defined Methods
- NDcPP22e:FPT_TUD_EXT.1: Trusted update
- VPNGW13:FPT_TUD_EXT.1: Trusted Update
- NDcPP22e:FTA_SSL_EXT.1: TSF-initiated Session Locking
- VPNGW13:FTA_VCM_EXT.1: VPN Client Management – per TD0656

5 Security Requirements

This section defines the Security Functional Requirements (SFRs) and Security Assurance Requirements (SARs) that serve to represent the security functional claims for the Target of Evaluation (TOE) and to scope the evaluation effort.

The SFRs have all been drawn from the NDcPP22e/ MACSEC10/ VPNGW13. The refinements and operations already performed in the NDcPP22e/ MACSEC10/ VPNGW13 are not identified (e.g., highlighted) here, rather the requirements have been copied from the NDcPP22e/ MACSEC10/ VPNGW13 and any residual operations have been completed herein. Of particular note, the NDcPP22e/ MACSEC10/ VPNGW13 made a number of refinements and completed some of the SFR operations defined in the Common Criteria (CC) and that PP should be consulted to identify those changes if necessary.

The SARs are also drawn from the NDcPP22e/ MACSEC10/ VPNGW13. The NDcPP22e/ MACSEC10/ VPNGW13 should be consulted for the assurance activity definitions.

5.1 TOE Security Functional Requirements

The following table identifies the SFRs that are satisfied by Wave Relay Devices TOE.

Requirement Class	Requirement Component
FAU: Security audit	NDcPP22e/MACSEC10:FAU_GEN.1: Audit Data Generation
	VPNGW13:FAU_GEN.1/VPN: Audit Data Generation (VPN Gateway)
	NDcPP22e:FAU_GEN.2: User identity association
	NDcPP22e:FAU_STG.1: Protected Audit Trail Storage
	NDcPP22e:FAU_STG_EXT.1: Protected Audit Event Storage
FCS: Cryptographic support	NDcPP22e:FCS_CKM.1: Cryptographic Key Generation
	VPNGW13:FCS_CKM.1/IKE: Cryptographic Key Generation (for IKE Peer Authentication)
	NDcPP22e:FCS_CKM.2: Cryptographic Key Establishment
	NDcPP22e:FCS_CKM.4: Cryptographic Key Destruction
	NDcPP22e:FCS_COP.1/DataEncryption: Cryptographic Operation (AES Data Encryption/Decryption)
	VPNGW13:FCS_COP.1/DataEncryption: Cryptographic Operation (AES Data Encryption/Decryption)
	MACSEC10:FCS_COP.1/MACSEC: Cryptographic Operation (MACsec AES Data Encryption/Decryption)
	NDcPP22e:FCS_COP.1/Hash: Cryptographic Operation (Hash Algorithm)
	NDcPP22e:FCS_COP.1/KeyedHash: Cryptographic Operation (Keyed Hash Algorithm)
	MACSEC10:FCS_COP.1/KeyedHashCMAC: Cryptographic Operation (AES-CMAC Keyed Hash Algorithm)
	NDcPP22e:FCS_COP.1/SigGen: Cryptographic Operation (Signature Generation and Verification)
	NDcPP22e:FCS_HTTPS_EXT.1: HTTPS Protocol
	NDcPP22e:FCS_IPSEC_EXT.1: IPsec Protocol
	VPNGW13:FCS_IPSEC_EXT.1: IPsec Protocol - per TD0657
	MACSEC10:FCS_MACSEC_EXT.1: MACsec
	MACSEC10:FCS_MACSEC_EXT.2: MACsec Integrity and Confidentiality
	MACSEC10:FCS_MACSEC_EXT.3: MACsec Randomness
	MACSEC10:FCS_MACSEC_EXT.4: MACsec Key Usage
	MACSEC10:FCS_MKA_EXT.1: MACsec Key Agreement
	NDcPP22e:FCS_RBG_EXT.1: Random Bit Generation
	NDcPP22e:FCS_TLSC_EXT.1: TLS Client Protocol Without Mutual Authentication - per TD0670
	NDcPP22e:FCS_TLSS_EXT.1: TLS Server Protocol Without Mutual Authentication - per TD0635

Requirement Class	Requirement Component
	NDcPP22e:FCS_TLSS_EXT.2
FIA: Identification and authentication	NDcPP22e:FIA_AFL.1: Authentication Failure Management
	MACSEC10:FIA_AFL.1: Authentication Attempt Limiting
	NDcPP22e:FIA_PMG_EXT.1: Password Management
	MACSEC10:FIA_PSK_EXT.1: Pre-Shared Key Composition
	VPNGW13:FIA_PSK_EXT.1: Pre-Shared Key Composition
	VPNGW13:FIA_PSK_EXT.2: Generated Pre-Shared Keys
	VPNGW13:FIA_PSK_EXT.3: Password-Based Pre-Shared Keys - per TD0771
	NDcPP22e:FIA_UAU.7: Protected Authentication Feedback
	NDcPP22e:FIA_UAU_EXT.2: Password-based Authentication Mechanism
	NDcPP22e:FIA_UIA_EXT.1: User Identification and Authentication
	NDcPP22e:FIA_X509_EXT.1/Rev: X.509 Certificate Validation
	VPNGW13:FIA_X509_EXT.1/Rev: X.509 Certificate Validation
	NDcPP22e:FIA_X509_EXT.2: X.509 Certificate Authentication
	VPNGW13:FIA_X509_EXT.2: X.509 Certificate Authentication
	NDcPP22e:FIA_X509_EXT.3: X.509 Certificate Requests
	VPNGW13:FIA_X509_EXT.3: X.509 Certificate Requests
FMT: Security management	NDcPP22e:FMT_MOF.1/ManualUpdate: Management of security functions behaviour
	NDcPP22e:FMT_MTD.1/CoreData: Management of TSF Data
	NDcPP22e:FMT_MTD.1/CryptoKeys: Management of TSF Data
	VPNGW13:FMT_MTD.1/CryptoKeys: Management of TSF Data
	NDcPP22e:FMT_SMF.1: Specification of Management Functions - per TD0631
	MACSEC10: FMT_SMF.1/MACSEC Specification of Management Functions – per TD0748
	VPNGW13:FMT_SMF.1/VPN: Specification of Management Functions
	NDcPP22e:FMT_SMR.2: Restrictions on Security Roles
FPF: Packet Filtering	VPNGW13:FPF_RUL_EXT.1: Packet Filtering Rules - per TD0683
FPT: Protection of the TSF	NDcPP22e:FPT_APW_EXT.1: Protection of Administrator Passwords
	MACSEC10:FPT_CAK_EXT.1: Protection of CAK Data
	MACSEC10:FPT_FLS.1: Failure with Preservation of Secure State
	VPNGW13:FPT_FLS.1/SelfTest: Failure with Preservation of Secure State (Self-Test Failures)
	MACSEC10:FPT_RPL.1: Replay Detection-per TD0746
	NDcPP22e:FPT_SKP_EXT.1: Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)
	NDcPP22e:FPT_STM_EXT.1: Reliable Time Stamps - per TD0632
	NDcPP22e:FPT_TST_EXT.1: TSF testing
	VPNGW13:FPT_TST_EXT.1: TSF Testing
	VPNGW13:FPT_TST_EXT.3: Self-Test with Defined Methods
	NDcPP22e:FPT_TUD_EXT.1: Trusted update
	VPNGW13:FPT_TUD_EXT.1: Trusted Update
FTA: TOE access	NDcPP22e:FTA_SSL.3: TSF-initiated Termination
	NDcPP22e:FTA_SSL.4: User-initiated Termination
	NDcPP22e:FTA_SSL_EXT.1: TSF-initiated Session Locking
	NDcPP22e:FTA_TAB.1: Default TOE Access Banners
	VPNGW13:FTA_TSE.1: TOE Session Establishment - per TD0656
	VPNGW13:FTA_VCM_EXT.1 VPN Client Management – per TD0656
FTP: Trusted path/channels	NDcPP22e:FTP_ITC.1: Inter-TSF trusted channel - per TD0639
	VPNGW13:FTP_ITC.1/VPN: Inter-TSF Trusted Channel (VPN Communications)
	MACSEC10:FTP_ITC.1/MACSEC: Inter-TSF Trusted Channel (MACsec Communications)
	NDcPP22e:FTP_TRP.1/Admin: Trusted Path - per TD0639

Requirement Class	Requirement Component
	MACSEC10:FTP_TRP.1/MACSEC: Trusted Path (MACsec Administration)

Table 5-1 TOE Security Functional Components

5.1.1 Security audit (FAU)

5.1.1.1 Audit Data Generation (NDcPP22e/MACSEC10:FAU_GEN.1)

NDcPP22e/MACSEC10:FAU_GEN.1.1

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

a Start-up and shut-down of the audit functions;
b All auditable events for the not specified level of audit; and
c All administrative actions comprising:
- Administrative login and logout (name of user account shall be logged if individual user accounts are required for administrators).
- Changes to TSF data related to configuration changes (in addition to the information that a change occurred it shall be logged what has been changed).
- Generating/import of, changing, or deleting of cryptographic keys (in addition to the action itself a unique key name or key reference shall be logged).
- Resetting passwords (name of related user account shall be logged).
- [***no other actions***];

d Specifically defined auditable events listed in Table 5-2.

NDcPP22e/MACSEC10: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 5-2.

Table 5-2 Audit Events from NDcPP22e and MACSEC10

Requirement	Audit Event	Additional Contents
NDcPP22e:FAU_GEN.1	None	None
NDcPP22e:FAU_GEN.2	None	None
NDcPP22e:FAU_STG_EXT.1	None	None
NDcPP22e:FCS_CKM.1	None	None
NDcPP22e:FCS_CKM.2	None	None
NDcPP22e:FCS_CKM.4	None	None
NDcPP22e:FCS_COP.1/DataEncryption	None	None
NDcPP22e:FCS_COP.1/Hash	None	None
NDcPP22e:FCS_COP.1/KeyedHash	None	None
NDcPP22e:FCS_COP.1/SigGen	None	None
NDcPP22e:FCS_HTTPS_EXT.1	Failure to establish a HTTPS Session.	Reason for failure.
NDcPP22e:FCS_IPSEC_EXT.1	Failure to establish an IPsec SA.	Reason for failure.
NDcPP22e:FCS_RBG_EXT.1	None	None
NDcPP22e:FCS_TLSC_EXT.1	Failure to establish a TLS Session.	Reason for failure.
NDcPP22e:FCS_TLSS_EXT.1	None	None
NDcPP22e:FCS_TLSS_EXT.2	Failure to authenticate the client.	Reason for failure.
NDcPP22e:FIA_AFL.1	Unsuccessful login attempt limit is met or exceeded.	Origin of the attempt (e.g., IP address).
NDcPP22e:FIA_PMG_EXT.1	None	None

Requirement	Audit Event	Additional Contents
NDcPP22e:FIA_UAU.7	None	None
NDcPP22e:FIA_UAU_EXT.2	All use of identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
NDcPP22e:FIA_UIA_EXT.1	All use of identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
NDcPP22e: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
NDcPP22e:FIA_X509_EXT.2	None	None
NDcPP22e:FIA_X509_EXT.3	None	None
NDcPP22e:FMT_MOF.1/ManualUpdate	Any attempt to initiate a manual update.	None
NDcPP22e:FMT_MTD.1/CoreData	None	None
NDcPP22e:FMT_MTD.1/CryptoKeys	None	None
NDcPP22e:FMT_SMF.1	All management activities of TSF data.	None
NDcPP22e:FMT_SMR.2	None	None
NDcPP22e:FPT_APW_EXT.1	None	None
NDcPP22e:FPT_ITT.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.
NDcPP22e:FPT_SKP_EXT.1	None	None
NDcPP22e: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).
NDcPP22e:FPT_TST_EXT.1	None	None
NDcPP22e:FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure).	None
NDcPP22e:FTA_SSL.3	The termination of a remote session by the session locking mechanism.	None
NDcPP22e:FTA_SSL.4	The termination of an interactive session.	None
NDcPP22e:FTA_SSL_EXT.1	(if 'lock the session' is selected) Any attempts at unlocking of an interactive session. (if 'terminate the session' is selected) The termination of a local session by the session locking mechanism.	None
NDcPP22e:FTA_TAB.1	None	None
NDcPP22e: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.

Requirement	Audit Event	Additional Contents
NDcPP22e:FTP_TRP.1/Admin	Initiation of the trusted path. Termination of the trusted path. Failure of the trusted path functions.	None
MACSEC10:FCS_MACSEC_EXT.1	Session establishment	Secure Channel Identifier (SCI)
MACSEC10:FCS_MACSEC_EXT.3	Creation and update of SAK	Creation and update times
MACSEC10:FCS_MACSEC_EXT.4	Creation of CA	Connectivity Association Key Names (CKNs)
MACSEC10:FPT_RPL.1	Detected replay attempt	None

5.1.1.2 Audit Data Generation (VPN Gateway) (VPNGW13:FAU_GEN.1/VPN)

VPNGW13:FAU_GEN.1.1/VPN

The TSF shall be able to generate an audit record of the following auditable events:
a. Start-up and shutdown of the audit functions
b. Indication that TSF self-test was completed
c. Failure of self-test
d. All auditable events for the not specified level of audit; and
e. auditable events defined in the Auditable Events for Mandatory Requirements table.

VPNGW13:FAU_GEN.1.2/VPN

The TSF shall record within each audit record at least the following information:
a. Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and
b. For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, additional information defined in the Auditable Events for Mandatory Requirements table for each auditable event, where applicable.

Table 5-3 Audit Events from VPNGW13

Requirement	Auditable Events	Additional Audit Record Content
VPNGW13:FAU_GEN.1/VPN	None	None
VPNGW13:FCS_CKM.1/IKE	None	None
VPNGW13:FCS_COP.1/DataEncryption	None	None
VPNGW13:FCS_EAP_EXT.1	None	None
VPNGW13:FCS_IPSEC_EXT.1	Failure to establish an IPsec SA.	Reason for failure
VPNGW13:FIA_HOTP_EXT.1	None	None
VPNGW13:FIA_PSK_EXT.1	None	None
VPNGW13:FIA_PSK_EXT.2	None	None
VPNGW13:FIA_PSK_EXT.3	None	None
VPNGW13:FIA_TOTP_EXT.1	None	None
VPNGW13:FIA_X509_EXT.1/Rev	None	None
VPNGW13:FIA_X509_EXT.2	None	None
VPNGW13:FIA_X509_EXT.3	None	None
VPNGW13:FMT_MTD.1/CryptoKeys	None	None
VPNGW13:FMT_SMF.1/VPN	All administrative actions	None
VPNGW13:FPF_MFA_EXT.1	None	None
VPNGW13:FPF_RUL_EXT.1	Application of rules configured with the 'log' operation	Source and destination addresses Source and destination ports Transport layer protocol

Requirement	Auditable Events	Additional Audit Record Content
VPNGW13:FPT_FLS.1/SelfTest	None	None
VPNGW13:FPT_TST_EXT.1	None	None
VPNGW13:FPT_TST_EXT.3	None	None
VPNGW13:FPT_TUD_EXT.1	None	None
VPNGW13:FTA_TSE.1	None	None
VPNGW13:FTA_VCM_EXT.1	None	None
VPNGW13:FTP_ITC.1/VPN	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 channel establishment attempt

5.1.1.3 User identity association (NDcPP22e:FAU_GEN.2)

NDcPP22e: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.

5.1.1.4 Protected Audit Trail Storage (NDcPP22e:FAU_STG.1)

NDcPP22e:FAU_STG.1.1
The TSF shall protect the stored audit records in the audit trail from unauthorised deletion.
NDcPP22e:FAU_STG.1.2
The TSF shall be able to prevent unauthorised modifications to the stored audit records in the audit trail.

5.1.1.5 Protected Audit Event Storage (NDcPP22e:FAU_STG_EXT.1)

NDcPP22e: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.
NDcPP22e: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.***]
NDcPP22e:FAU_STG_EXT.1.3
The TSF shall [***overwrite previous audit records according to the following rule: [audit record will overwrite the oldest audit record]***] when the local storage space for audit data is full.

5.1.2 Cryptographic support (FCS)

5.1.2.1 Cryptographic Key Generation (NDcPP22e:FCS_CKM.1)

NDcPP22e:FCS_CKM.1.1
The TSF shall generate asymmetric cryptographic keys in accordance with a specified cryptographic key generation algorithm: [
- ***ECC schemes using 'NIST curves' [P-256, P-384] that meet the following: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Appendix B.***

].

5.1.2.2 Cryptographic Key Generation (for IKE Peer Authentication) (VPNGW13:FCS_CKM.1/IKE)

VPNGW13:FCS_CKM.1.1/IKE
The TSF shall generate asymmetric cryptographic keys used for IKE peer authentication in accordance with a specified cryptographic key generation algorithm: [

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

and [

- ***no other key generation algorithm***

]

and specified cryptographic key sizes equivalent to, or greater than, a symmetric key strength of 112 bits. (TD0878 applied)

5.1.2.3 Cryptographic Key Establishment (NDcPP22e:FCS_CKM.2)

NDcPP22e:FCS_CKM.2.1

The TSF shall perform cryptographic key establishment in accordance with a specified cryptographic key establishment method: [

- ***Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 3, 'Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography' (TD0581 applied)***

].

5.1.2.4 Cryptographic Key Destruction (NDcPP22e:FCS_CKM.4)

NDcPP22e: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.

5.1.2.5 Cryptographic Operation (AES Data Encryption/Decryption) (NDcPP22e:FCS_COP.1/DataEncryption)

NDcPP22e:FCS_COP.1.1/DataEncryption

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

5.1.2.6 Cryptographic Operation (AES Data Encryption/Decryption) (VPNGW13:FCS_COP.1/DataEncryption)

VPNGW13:FCS_COP.1.1/DataEncryption

The TSF shall perform encryption/decryption in accordance with a specified cryptographic algorithm AES used in [***CBC, GCM***] and [***no other***] mode and cryptographic key sizes [***256 bits***] , and [***no other cryptographic key sizes***] that meet the following: AES as specified in ISO 18033-3, [***CBC as specified in ISO 10116, GCM as specified in ISO 19772***] and [***no other standards***].

5.1.2.7 Cryptographic Operation (MACsec AES Data Encryption/Decryption) (MACSEC10:FCS_COP.1/MACSEC)

MACSEC10:FCS_COP.1.1/MACSEC

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. (TD0466 applied, supersedes TD0134 and TD0357)

5.1.2.8 Cryptographic Operation (Hash Algorithm) (NDcPP22e:FCS_COP.1/Hash)

NDcPP22e:FCS_COP.1.1/Hash

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

5.1.2.9 Cryptographic Operation (Keyed Hash Algorithm) (NDcPP22e:FCS_COP.1/KeyedHash)

NDcPP22e:FCS_COP.1.1/KeyedHash

The TSF shall perform keyed-hash message authentication in accordance with a specified cryptographic algorithm [***HMAC-SHA-384***] and cryptographic key sizes [***384***] and message digest sizes [***384, 512***] bits that meet the following: ISO/IEC 9797-2:2011, Section 7 'MAC Algorithm 2'.

5.1.2.10 Cryptographic Operation (AES-CMAC Keyed Hash Algorithm) (MACSEC10:FCS_COP.1/KeyedHashCMAC)

MACSEC10:FCS_COP.1.1/KeyedHashCMAC

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.

5.1.2.11 Cryptographic Operation (Signature Generation and Verification) (NDcPP22e:FCS_COP.1/SigGen)

NDcPP22e: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],***
- ***Elliptic Curve Digital Signature Algorithm and cryptographic key sizes [256 bits, 384 bits]***]

that meet the following: [

- ***For RSA schemes: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSA-PKCS1v1_5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3,***
- ***For ECDSA schemes: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Section 6 and Appendix D, Implementing 'NIST curves' [P-256, P-384]; ISO/IEC 14888-3, Section 6.4***].

5.1.2.12 NTP Protocol (NDcPP22e:FCS_NTP_EXT.1)

NDcPP22e:FCS_NTP_EXT.1.1

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

NDcPP22e:FCS_NTP_EXT.1.2

The TSF shall update its system time using [***Authentication using [SHA1, SHA256, SHA384, SHA512, AES-CBC-128, AES-CBC-256] as the message digest algorithm(s);***].

NDcPP22e:FCS_NTP_EXT.1.3

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

NDcPP22e:FCS_NTP_EXT.1.4

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

5.1.2.13 Random Bit Generation (NDcPP22e:FCS_RBG_EXT.1)

NDcPP22e:FCS_RBG_EXT.1.1

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

NDcPP22e: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 18031:2011Table C.1 'Security Strength Table for Hash Functions', of the keys and hashes that it will generate.

5.1.2.14 HTTPS Protocol (NDcPP22e:FCS_HTTPS_EXT.1)

NDcPP22e:FCS_HTTPS_EXT.1.1
The TSF shall implement the HTTPS protocol that complies with RFC 2818.

NDcPP22e:FCS_HTTPS_EXT.1.2
The TSF shall implement HTTPS using TLS.

NDcPP22e:FCS_HTTPS_EXT.1.3
If a peer certificate is presented, the TSF shall [***not require client authentication***] if the peer certificate is deemed invalid.

5.1.2.15 IPsec Protocol - (NDcPP22e:FCS_IPSEC_EXT.1)

NDcPP22e:FCS_IPSEC_EXT.1.1
The TSF shall implement the IPsec architecture as specified in RFC 4301.

NDcPP22e:FCS_IPSEC_EXT.1.2
The TSF shall have a nominal, final entry in the SPD that matches anything that is otherwise unmatched, and discards it.

NDcPP22e:FCS_IPSEC_EXT.1.3
The TSF shall implement [***tunnel mode***].

NDcPP22e:FCS_IPSEC_EXT.1.4
The TSF shall implement the IPsec protocol ESP as defined by RFC 4303 using the cryptographic algorithms [***AES-CBC-256 (RFC 3602), AES-GCM-256 (RFC 4106)***] together with a Secure Hash Algorithm (SHA)-based HMAC [***HMAC-SHA-384***].

NDcPP22e:FCS_IPSEC_EXT.1.5
The TSF shall implement the protocol: [
- ***IKEv2 as defined in RFC 5996 and [with mandatory support for NAT traversal as specified in RFC 5996, section 2.23], and [no other RFCs for hash functions]***].

NDcPP22e:FCS_IPSEC_EXT.1.6
The TSF shall ensure the encrypted payload in the [***IKEv2***] protocol uses the cryptographic algorithms [***AES-CBC-256 (specified in RFC 3602), AES-GCM-256 (specified in RFC 5282)***].

NDcPP22e:FCS_IPSEC_EXT.1.7
The TSF shall ensure that [
- ***IKEv2 SA lifetimes can be configured by a Security Administrator based on [length of time, where the time values can be configured within [24] hours]***].

NDcPP22e:FCS_IPSEC_EXT.1.8
The TSF shall ensure that [
- ***IKEv2 Child SA lifetimes can be configured by a Security Administrator based on [length of time, where the time values can be configured within [8] hours]].***

NDcPP22e:FCS_IPSEC_EXT.1.9
The TSF shall generate the secret value x used in the IKE Diffie-Hellman key exchange ('x' in g^x mod p) using the random bit generator specified in FCS_RBG_EXT.1, and having a length of at least [**384 (group 20)**] bits.

NDcPP22e:FCS_IPSEC_EXT.1.10
The TSF shall generate nonces used in [***IKEv2***] exchanges of length [***according to the security strength associated with the negotiated Diffie-Hellman group***].

NDcPP22e:FCS_IPSEC_EXT.1.11
The TSF shall ensure that IKE protocols implement DH Group(s) [***[19 (256-bit Random ECP), 20 (384-bit Random ECP)]***].

NDcPP22e:FCS_IPSEC_EXT.1.12
The TSF shall be able to ensure by default that the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the [***IKEv2 IKE_SA***] connection is greater

than or equal to the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the [***IKEv2 CHILD_SA***] connection.

NDcPP22e:FCS_IPSEC_EXT.1.13

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

NDcPP22e:FCS_IPSEC_EXT.1.14

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

5.1.2.16 IPsec Protocol - per TD0824 (VPNGW13:FCS_IPSEC_EXT.1)

VPNGW13:FCS_IPSEC_EXT.1.1

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

VPNGW13:FCS_IPSEC_EXT.1.2

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

VPNGW13:FCS_IPSEC_EXT.1.3

The TSF shall implement [***tunnel mode***].

VPNGW13:FCS_IPSEC_EXT.1.4

The TSF shall implement the IPsec protocol ESP as defined by RFC 4303 using the cryptographic algorithms [***AES-CBC-256 (RFC 3602), AES-GCM-256 (specified in RFC 4106)***] and [***no other algorithm***] together with a Secure Hash Algorithm (SHA)-based HMAC [***HMAC-SHA-384***].

VPNGW13:FCS_IPSEC_EXT.1.5

The TSF shall implement the protocol: [***IKEv2 as defined in RFC 7296 and (choose one of): [with mandatory support for NAT traversal as specified in RFC 7296, section 2.23] and [no other RFCs for hash functions]***]. (TD0824 applied)

VPNGW13:FCS_IPSEC_EXT.1.6

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

VPNGW13:FCS_IPSEC_EXT.1.7

The TSF shall ensure that [

- ***IKEv2 SA lifetimes can be configured by a Security Administrator based on [length of time, where the time values can be configured within [24] hours]***].

VPNGW13:FCS_IPSEC_EXT.1.8

The TSF shall ensure that [

- ***IKEv2 Child SA lifetimes can be configured by a Security Administrator based on [length of time, where the time values can be configured within [8] hours]].***

VPNGW13:FCS_IPSEC_EXT.1.9

The TSF shall generate the secret value x used in the IKE Diffie-Hellman key exchange ('x' in g^x mod p) using the random bit generator specified in FCS_RBG_EXT.1, and having a length of at least [**384 (group 20)**] bits.

VPNGW13:FCS_IPSEC_EXT.1.10

The TSF shall generate nonces used in [***IKEv2***] exchanges of length [***according to the security strength associated with the negotiated DH group, at least 128 bits in size and at least half the output size of the negotiated pseudorandom function (PRF) hash***].

VPNGW13:FCS_IPSEC_EXT.1.11

The TSF shall ensure that IKE protocols implement DH Groups

- 19 (256-bit Random ECP),
- 20 (384-bit Random ECP) according to RFC 5114 and
- [***no other DH Groups] according to RFC 5114***].

VPNGW13:FCS_IPSEC_EXT.1.12

The TSF shall be able to ensure by default that the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the [***IKEv2 IKE_SA***] connection is greater

than or equal to the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the [***IKEv2 CHILD_SA***] connection.

VPNGW13:FCS_IPSEC_EXT.1.13

The TSF shall ensure that [***IKEv2***] protocols perform peer authentication using [***ECDSA***] that use X.509v3 certificates that conform to RFC 4945 and [***Pre-shared Keys that conform to RFC 8784***]. (TD0824 applied)

VPNGW13:FCS_IPSEC_EXT.1.14

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

5.1.2.17 MACsec (MACSEC10:FCS_MACSEC_EXT.1)

MACSEC10:FCS_MACSEC_EXT.1.1

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

MACSEC10:FCS_MACSEC_EXT.1.2

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

MACSEC10: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.

MACSEC10:FCS_MACSEC_EXT.1.4

The TSF shall permit only EAPOL (Port Access Entity (PAE) EtherType 88-8E), MACsec frames (EtherType 88-E5), and MAC control frames (EtherType is 88-08) and shall discard others

5.1.2.18 MACsec Integrity and Confidentiality (MACSEC10:FCS_MACSEC_EXT.2)

MACSEC10:FCS_MACSEC_EXT.2.1

The TOE shall implement MACsec with support for integrity protection with a confidentiality offset of [***0***].

MACSEC10:FCS_MACSEC_EXT.2.2

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

MACSEC10:FCS_MACSEC_EXT.2.3

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

5.1.2.19 MACsec Randomness (MACSEC10:FCS_MACSEC_EXT.3)

MACSEC10: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.

MACSEC10: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.

5.1.2.20 MACsec Key Usage (MACSEC10:FCS_MACSEC_EXT.4)

MACSEC10:FCS_MACSEC_EXT.4.1

The TSF shall support peer authentication using pre-shared keys (PSK) [***no other methods***].

MACSEC10:FCS_MACSEC_EXT.4.2

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

MACSEC10:FCS_MACSEC_EXT.4.3

The TSF shall support specifying a lifetime for CAKs.

MACSEC10:FCS_MACSEC_EXT.4.4
The TSF shall associate Connectivity Association Key Names (CKNs) with SAKs that are defined by the KDF using the CAK as input data (per 802.1X, section 9.8.1).

MACSEC10:FCS_MACSEC_EXT.4.5
The TSF shall associate CKNs with CAKs. The length of the CKN shall be an integer number of octets, between 1 and 32 (inclusive).

5.1.2.21 MACsec Key Agreement (MACSEC10:FCS_MKA_EXT.1)

MACSEC10:FCS_MKA_EXT.1.1
The TSF shall implement Key Agreement Protocol (MKA) in accordance with IEEE 802.1X-2010 and 802.1Xbx-2014.

MACSEC10:FCS_MKA_EXT.1.2
The TSF shall provide assurance of the integrity of MKA protocol data units (MKPDUs) using an Integrity Check Value (ICV) derived from an Integrity Check Value Key (ICK).

MACSEC10:FCS_MKA_EXT.1.3
The TSF shall provide the ability to derive an Integrity Check Value Key (ICK) from a CAK using a KDF.

MACSEC10:FCS_MKA_EXT.1.4
The TSF shall enforce an MKA Lifetime Timeout limit of 6.0 seconds and MKA Bounded Hello Time limit of 0.5 seconds.

MACSEC10:FCS_MKA_EXT.1.5
The Key Server shall refresh a SAK when it expires. The Key Server shall distribute a SAK by [***pairwise CAKs that are PSKs***].

MACSEC10:FCS_MKA_EXT.1.6
The Key Server shall distribute a fresh SAK whenever a member is added to or removed from the live membership of the CA.

MACSEC10:FCS_MKA_EXT.1.7
The TSF shall validate MKPDUs according to IEEE 802.1X-2010 Section 11.11.2. In particular, the TSF shall discard without further processing any MKPDUs to which any of the following conditions apply:

a. The destination address of the MKPDU was an individual address
b. The MKPDU is less than 32 octets long
c. The MKPDU comprises fewer octets than indicated by the Basic Parameter Set body length, as encoded in bits 4 through 1 of octet 3 and bits 8 through 1 of octet 4, plus 16 octets of ICV
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-2010 Section 9.4.1.
b. If the Algorithm Agility parameter is unrecognized or not implemented by the receiver, its value can be recorded for diagnosis but the received MKPDU shall be discarded without further processing.

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

5.1.2.22 TLS Client Protocol Without Mutual Authentication (NDcPP22e:FCS_TLSC_EXT.1)

NDcPP22e: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_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289***

] and no other ciphersuites.

NDcPP22e:FCS_TLSC_EXT.1.2
The TSF shall verify that the presented identifier matches [***IPv4 address in CN or SAN***].

NDcPP22e: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***].

NDcPP22e:FCS_TLSC_EXT.1.4
The TSF shall [***present the Supported Elliptic Curves/Supported Groups Extension with the following curves/groups: [secp384r1] and no other curves/groups***] in the Client Hello.

5.1.2.23 TLS Server Protocol Without Mutual Authentication - per TD0635 (NDcPP22e:FCS_TLSS_EXT.1)

NDcPP22e:FCS_TLSS_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_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289***

] and no other ciphersuites.

NDcPP22e:FCS_TLSS_EXT.1.2
The TSF shall deny connections from clients requesting SSL 2.0, SSL 3.0, TLS 1.0 and [***TLS 1.1***].

NDcPP22e:FCS_TLSS_EXT.1.3
The TSF shall perform key establishment for TLS using [***ECDHE curves [secp384r1] and no other curves***].

NDcPP22e:FCS_TLSS_EXT.1.4
The TSF shall support [***no session resumption or session tickets***].

5.1.3 Identification and authentication (FIA)

5.1.3.1 Authentication Failure Management (NDcPP22e:FIA_AFL.1)

NDcPP22e:FIA_AFL.1.1
The TSF shall detect when an Administrator configurable positive integer within [**3, 5, or 10**] unsuccessful authentication attempts occur related to Administrators attempting to authenticate remotely using a password.

NDcPP22e:FIA_AFL.1.2
When the defined number of unsuccessful authentication attempts has been met, the TSF shall [***prevent the offending Administrator from successfully establishing a remote session using any authentication method that involves a password until an Administrator defined time period has elapsed***].

5.1.3.2 Password Management (NDcPP22e:FIA_PMG_EXT.1)

NDcPP22e: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 [**8**] and [**256**] characters.

5.1.3.3 Extended: Pre-Shared Key Composition (MACsecEP12:FIA_PSK_EXT.1)

MACSEC10:FIA_PSK_EXT.1.1
The TSF shall use PSKs for MKA as defined by IEEE 802.1X-2010, [***no other protocols***].

MACSEC10:FIA_PSK_EXT.1.2
The TSF shall be able to [***generate using the random bit generator specified in FCS_RBG_EXT.1***] bit-based PSKs.

5.1.3.4 Pre-Shared Key Composition (VPNGW13:FIA_PSK_EXT.1)

VPNGW13:FIA_PSK_EXT.1.1
The TSF shall be able to use pre-shared keys for IPsec and [***IKEv2***].

VPNGW13:FIA_PSK_EXT.1.2
The TSF shall be able to accept the following as pre-shared keys: [
- ***generated bit-based,***] keys.

5.1.3.5 Generated Pre-Shared Keys (VPNGW13:FIA_PSK_EXT.2)

VPNGW13:FIA_PSK_EXT.2.1
The TSF shall be able to [
- ***accept externally generated pre-shared keys***].

5.1.3.6 Protected Authentication Feedback (NDcPP22e:FIA_UAU.7)

NDcPP22e: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.

5.1.3.7 Password-based Authentication Mechanism (NDcPP22e:FIA_UAU_EXT.2)

NDcPP22e:FIA_UAU_EXT.2.1
The TSF shall provide a local [***password-based***] authentication mechanism to perform local administrative user authentication.

5.1.3.8 User Identification and Authentication (NDcPP22e:FIA_UIA_EXT.1)

NDcPP22e: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;
- [Display the Node's name and IP address
- Display the Node's battery level
- Display Status via LED
- Provide Rotary knob for Zeroization (Cryptographic Keys, Management Password, HTTPS certificate)
- Change Talkgroups
- Modify Audio Volume].

NDcPP22e:FIA_UIA_EXT.1.2
The TSF shall require each administrative user to be successfully identified and authenticated before allowing any other TSF-mediated actions on behalf of that administrative user.

5.1.3.9 X.509 Certificate Validation (NDcPP22e:FIA_X509_EXT.1/Rev)

NDcPP22e:FIA_X509_EXT.1.1/Rev
The TSF shall validate certificates in accordance with the following rules:
- RFC 5280 certificate validation and certification path validation supporting a minimum path length of three certificates.
- The certification path must terminate with a trusted CA certificate designated as a trust anchor.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of the certificate using [
  - ***the Online Certificate Status Protocol (OCSP) as specified in RFC 6960,***
    - ***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.

NDcPP22e: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.

5.1.3.10 X.509 Certificate Validation (VPNGW13:FIA_X509_EXT.1/Rev)

VPNGW13:FIA_X509_EXT.1.1/Rev

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

- RFC 5280 certificate validation and certification path validation supporting a minimum path length of three certificates.
- The certification path must terminate with a trusted CA certificate designated as a trust anchor.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of the certificate using [
  - ***a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3, and***
  - ***Certificate Revocation List (CRL) as specified in RFC 8603 Section 7***]. [1]
- 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.

5.1.3.11 X.509 Certificate Authentication (NDcPP22e:FIA_X509_EXT.2)

NDcPP22e:FIA_X509_EXT.2.1

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

NDcPP22e: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***].

5.1.3.12 X.509 Certificate Authentication (VPNGW13:FIA_X509_EXT.2)

NDcPP22e:FIA_X509_EXT.2.1

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

NDcPP22e: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***].

[1] RFC 8603 is required for compliance with CSfC VPNGW13 Selections documents. This delta has been allowed by NIAP for other evaluations.

5.1.3.13 X.509 Certificate Requests (NDcPP22e:FIA_X509_EXT.3)

NDcPP22e:FIA_X509_EXT.3.1
The TSF shall generate a Certification Request as specified by RFC 2986 and be able to provide the following information in the request: public key and [***Common Name***].

NDcPP22e:FIA_X509_EXT.3.2
The TSF shall validate the chain of certificates from the Root CA upon receiving the CA Certificate Response.

5.1.3.14 X.509 Certificate Requests (VPNGW13:FIA_X509_EXT.3)

VPNGW13: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***].

VPNGW13:FIA_X509_EXT.3.2
The TSF shall validate the chain of certificates from the Root CA upon receiving the CA Certificate Response.

5.1.4 Security management (FMT)

5.1.4.1 Management of security functions behavior (NDcPP22e:FMT_MOF.1/ManualUpdate)

NDcPP22e:FMT_MOF.1.1/ManualUpdate
The TSF shall restrict the ability to enable the functions to perform manual updates to Security Administrators.

5.1.4.2 Management of Security Functions Behaviour (NDcPP22e:FMT_MOF.1/Services)

NDcPP22e:FMT_MOF.1.1/Services
The TSF shall restrict the ability to start and stop services to Security Administrators.

5.1.4.3 Management of TSF Data (NDcPP22e:FMT_MTD.1/CoreData)

NDcPP22e:FMT_MTD.1.1/CoreData
The TSF shall restrict the ability to manage the TSF data to Security Administrators.

5.1.4.4 Management of TSF Data (NDcPP22e:FMT_MTD.1/CryptoKeys)

NDcPP22e:FMT_MTD.1.1/CryptoKeys
The TSF shall restrict the ability to manage the cryptographic keys to Security Administrators.

5.1.4.5 Management of TSF Data (VPNGW13:FMT_MTD.1/CryptoKeys)

VPNGW13:FMT_MTD.1.1/CryptoKeys
The TSF shall restrict the ability to manage the cryptographic keys and certificates used for VPN operation to Security Administrators.

5.1.4.6 Specification of Management Functions - per TD0631 (NDcPP22e:FMT_SMF.1)

NDcPP22e:FMT_SMF.1.1
The TSF shall be capable of performing the following management functions:
- Ability to administer the TOE locally and remotely;
- Ability to configure the access banner;
- Ability to configure the session inactivity time before session termination or locking;
- Ability to update the TOE, and to verify the updates using [***digital signature***] capability prior to installing those updates;
- Ability to configure the authentication failure parameters for FIA_AFL.1;
- [***Ability to start and stop services;***

- ***Ability to modify the behavior of the transmission of audit data to an external IT entity,***
- ***Ability to manage the cryptographic keys,***
- ***Ability to configure the cryptographic functionality,***
- ***Ability to configure the lifetime for IPsec SAs,***
- ***Ability to set the time which is used for time-stamps,***
- ***Ability to configure NTP,***
- ***Ability to configure the reference identifier for the peer,***
- ***Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors,***
- ***Ability to import X509v3 certificates to the TOE's trust store***].

5.1.4.7 Specification of Management Functions (VPNGW13:FMT_SMF.1/VPN)

VPNGW13:FMT_SMF.1.1/VPN

The TSF shall be capable of performing the following management functions:
- Definition of packet filtering rules;
- Association of packet filtering rules to network interfaces;
- Ordering of packet filtering rules by priority;
- [***No other capabilities***].

5.1.4.8 Specification of Management Functions (MACsec) (MACSEC10:FMT_SMF.1/MACSEC)-per TD0748

FMT_SMF.1/MACSEC.1:

The TSF shall be capable of performing the following management functions related to MACsec functionality: Ability of a Security Administrator to:
- Manage a PSK-based CAK and install it in the device
- Manage the Key Server to create, delete, and activate MKA participants ***[as specified in 802.1X-2020, sections 9.13 and 9.16 (cf. MIB object ieee8021XKayMkaParticipantEntry) and section.12.2 (cf. function createMKA())]***
- Specify a lifetime of a CAK
- Enable, disable, or delete a PSK-based CAK using ***[the MIB object ieee8021XKayMkaPartActivateControl]***
- ***[No other management functions].***

5.1.4.9 Restrictions on Security Roles (NDcPP22e:FMT_SMR.2)

NDcPP22e:FMT_SMR.2.1

The TSF shall maintain the roles: - Security Administrator.

NDcPP22e:FMT_SMR.2.2

The TSF shall be able to associate users with roles.

NDcPP22e: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.

5.1.5 Packet Filtering (FPF)

5.1.5.1 Packet Filtering Rules (VPNGW13:FPF_RUL_EXT.1)

VPNGW13:FPF_RUL_EXT.1.1

The TSF shall perform Packet Filtering on network packets processed by the TOE.

VPNGW13:FPF_RUL_EXT.1.2

The TSF shall allow the definition of Packet Filtering rules using the following network protocols and protocol fields:
- IPv4 (RFC 791)

    - source address
    - destination address
    - protocol
  - IPv6 (RFC 8200)
    - source address
    - destination address
    - next header (protocol)
  - TCP (RFC 793)
    - source port
    - destination port
  - UDP (RFC 768)
    - source port
    - destination port.

VPNGW13:FPF_RUL_EXT.1.3
The TSF shall allow the following operations to be associated with Packet Filtering rules: permit and drop with the capability to log the operation.

VPNGW13:FPF_RUL_EXT.1.4
The TSF shall allow the Packet Filtering rules to be assigned to each distinct network interface.

VPNGW13:FPF_RUL_EXT.1.5
The TSF shall process the applicable Packet Filtering rules (as determined in accordance with VPNGW13:FPF_RUL_EXT.1.4) in the following order: Administrator-defined.

VPNGW13:FPF_RUL_EXT.1.6
The TSF shall drop traffic if a matching rule is not identified.

5.1.6 Protection of the TSF (FPT)

5.1.6.1 Protection of Administrator Passwords (NDcPP22e:FPT_APW_EXT.1)

NDcPP22e:FPT_APW_EXT.1.1
The TSF shall store administrative passwords in non-plaintext form.

NDcPP22e:FPT_APW_EXT.1.2
The TSF shall prevent the reading of plaintext administrative passwords.

5.1.6.2 Protection of CAK Data (MACSEC10:FPT_CAK_EXT.1)

MACSEC10:FPT_CAK_EXT.1.1
The TSF shall prevent reading of CAK values by administrators.

5.1.6.3 Failure with Preservation of Secure State (MACSEC10:FPT_FLS.1)

MACSEC10:FPT_FLS.1.1
The TSF shall fail-secure when any of the following types of failures occur: failure of the power-on self-tests, failure of integrity check of the TSF executable image, failure of noise source health tests.

5.1.6.4 Failure with Preservation of Secure State (Self-Test Failures) (VPNGW13:FPT_FLS.1/SelfTest)

VPNGW13:FPT_FLS.1.1/SelfTest
The TSF shall shut down when the following types of failures occur: failure of the power-on self-tests, failure of integrity check of the TSF executable image, failure of noise source health tests.

5.1.6.5 Replay Detection - per TD0746 (MACSEC10:FPT_RPL.1)

MACSEC10:FPT_RPL.1.1
The TSF shall detect replay for the following entities: MPDUs, MKA frames.

MACSEC10:FPT_RPL.1.2
The TSF shall perform discarding of the replayed data, logging of the detected replay attempt when replay is detected.

5.1.6.6 Protection of TSF Data (for reading of all pre-shared, symmetric and private keys) (NDcPP22e:FPT_SKP_EXT.1)

NDcPP22e:FPT_SKP_EXT.1.1
The TSF shall prevent reading of all pre-shared keys, symmetric keys, and private keys.

5.1.6.7 Reliable Time Stamps - per TD0632 (NDcPP22e:FPT_STM_EXT.1)

NDcPP22e:FPT_STM_EXT.1.1
The TSF shall be able to provide reliable time stamps for its own use.
NDcPP22e:FPT_STM_EXT.1.2
The TSF shall [***allow the Security Administrator to set the time, synchronise time with an NTP server***].

5.1.6.8 TSF testing (NDcPP22e:FPT_TST_EXT.1)

NDcPP22e:FPT_TST_EXT.1.1
The TSF shall run a suite of the following self-tests [***during initial start-up (on power on)]***] to demonstrate the correct operation of the TSF: [***algorithm KAT tests, TSF integrity tests***].

5.1.6.9 TSF Testing - per TD0824 (VPNGW13:FPT_TST_EXT.1)

VPNGW13:FPT_TST_EXT.1.1
The TSF shall run a suite of the following self-tests:
- During initial start-up (on power on) to verify the integrity of the TOE firmware and software;
- Prior to providing any cryptographic service and [***continuously***]
- [no other]

to demonstrate the correct operation of the TSF: noise source health tests. (TD0824 applied)
VPNGW13:FPT_TST_EXT.1.2
The TSF shall respond to [***all failures,***] by [***entering a maintenance mode***]. (per TD0824)

5.1.6.10 Self-Test with Defined Methods (VPNGW13:FPT_TST_EXT.3)

VPNGW13:FPT_TST_EXT.3.1
The TSF shall run a suite of the following self-tests when loaded for execution to demonstrate the correct operation of the TSF: integrity verification of stored executable code.
VPNGW13:FPT_TST_EXT.3.2
The TSF shall execute the self-testing through a TSF-provided cryptographic service specified in FCS_COP.1/SigGen.

5.1.6.11 Trusted update (NDcPP22e:FPT_TUD_EXT.1)

NDcPP22e: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***].
NDcPP22e: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***].
NDcPP22e:FPT_TUD_EXT.1.3
The TSF shall provide means to authenticate firmware/software updates to the TOE using a [***digital signature***] prior to installing those updates.

5.1.6.12 Trusted Update - per TD0824 (VPNGW13:FPT_TUD_EXT.1)

VPNGW13: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***].

VPNGW13: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***].

VPNGW13:FPT_TUD_EXT.1.3
The TSF shall provide means to authenticate firmware/software updates to the TOE using a digital signature mechanism and [***no other mechanisms***] prior to installing those updates.

5.1.7 TOE access (FTA)

5.1.7.1 TSF-initiated Termination (NDcPP22e:FTA_SSL.3)

NDcPP22e:FTA_SSL.3.1
The TSF shall terminate a remote interactive session after a Security Administrator-configurable time interval of session inactivity.

5.1.7.2 User-initiated Termination (NDcPP22e:FTA_SSL.4)

NDcPP22e:FTA_SSL.4.1
The TSF shall allow Administrator-initiated termination of the Administrator's own interactive session.

5.1.7.3 TSF-initiated Session Locking (NDcPP22e:FTA_SSL_EXT.1)

NDcPP22e:FTA_SSL_EXT.1.1
The TSF shall, for local interactive sessions, [***terminate the session***] after a Security Administrator-specified time period of inactivity.

5.1.7.4 Default TOE Access Banners (NDcPP22e:FTA_TAB.1)

NDcPP22e: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.

5.1.8 Trusted path/channels (FTP)

5.1.8.1 Inter-TSF trusted channel - per TD0639 (NDcPP22e:FTP_ITC.1)

NDcPP22e:FTP_ITC.1.1
The TSF shall be capable of using [***IPsec, TLS***] to provide a trusted communication channel between itself and authorized IT entities supporting the following capabilities: audit server, [***remote VPN gateways or peers, controlled network devices***] 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.

NDcPP22e:FTP_ITC.1.2
The TSF shall permit the TSF or the authorized IT entities to initiate communication via the trusted channel.

NDcPP22e:FTP_ITC.1.3
The TSF shall initiate communication via the trusted channel for [**transmitting audit records to an audit server, remote VPN gateways or peers, and communication with a controlled network device**].

5.1.8.2 Inter-TSF Trusted Channel (MACsec Communications) (MACSEC10:FTP_ITC.1/MACSEC)

MACSEC10:FTP_ITC.1.1/MACSEC
The TSF shall provide a communication channel between itself and a MACsec peer that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.

MACSEC10:FTP_ITC.1.2/MACSEC
The TSF shall permit [***the TSF***] to initiate communication via the trusted channel.

MACSEC10:FTP_ITC.1.3/MACSEC
The TSF shall initiate communication via the trusted channel for communications with MACsec peers that require the use of MACsec.

5.1.8.3 Inter-TSF Trusted Channel (VPN Communications) (VPNGW13:FTP_ITC.1/VPN)

VPNGW13:FTP_ITC.1.1/VPN
The TSF shall be capable of using IPsec to provide a communication channel between itself and authorized IT entities supporting VPN communications that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from disclosure and detection of modification of the channel data.

VPNGW13:FTP_ITC.1.2/VPN
The TSF shall permit the authorized IT entities to initiate communication via the trusted channel.

VPNGW13:FTP_ITC.1.3/VPN
The TSF shall initiate communication via the trusted channel for (choose one of): [***remote VPN gateways or peers***].

5.1.8.4 Trusted Path - per TD0639 (NDcPP22e:FTP_TRP.1/Admin)

NDcPP22e:FTP_TRP.1.1/Admin
The TSF shall be capable of using [***TLS, HTTPS***] 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.

NDcPP22e:FTP_TRP.1.2/Admin
The TSF shall permit remote Administrators to initiate communication via the trusted path.

NDcPP22e:FTP_TRP.1.3/Admin
The TSF shall require the use of the trusted path for initial Administrator authentication and all remote administration actions.

5.1.8.5 Trusted Path (MACsec Administration) (MACSEC10:FTP_TRP.1/MACSEC)

MACSEC10:FTP_TRP.1.1/MACSEC
The TSF shall provide a communication path between itself and remote users using [***MACsec,***] that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from modification, disclosure.

MACSEC10:FTP_TRP.1.2/MACSEC
The TSF shall permit remote users to initiate communication via the trusted path.

MACSEC10:FTP_TRP.1.3/MACSEC
The TSF shall require the use of the trusted path for remote administration of MACsec management functions as defined in FMT_SMF.1/MACSEC.

5.2 TOE Security Assurance Requirements

The SARs for the TOE are the components as specified in Part 3 of the Common Criteria. Note that the SARs have effectively been refined with the assurance activities explicitly defined in association with both the SFRs and SARs.

Requirement Class	Requirement Component
ADV: Development	ADV_FSP.1: Basic Functional Specification

AGD: Guidance documents	AGD_OPE.1: Operational User Guidance
	AGD_PRE.1: Preparative Procedures
ALC: Life-cycle support	ALC_CMC.1: Labelling of the TOE
	ALC_CMS.1: TOE CM Coverage
ATE: Tests	ATE_IND.1: Independent Testing - Conformance
AVA: Vulnerability assessment	AVA_VAN.1: Vulnerability Survey

Table 5-4 Assurance Components

5.2.1 Development (ADV)

5.2.1.1 Basic Functional Specification (ADV_FSP.1)

ADV_FSP.1.1d
The developer shall provide a functional specification.

ADV_FSP.1.2d
The developer shall provide a tracing from the functional specification to the SFRs.

ADV_FSP.1.1c
The functional specification shall describe the purpose and method of use for each SFR-enforcing and SFR-supporting TSFI.

ADV_FSP.1.2c
The functional specification shall identify all parameters associated with each SFR-enforcing and SFR-supporting TSFI.

ADV_FSP.1.3c
The functional specification shall provide rationale for the implicit categorization of interfaces as SFR-non-interfering.

ADV_FSP.1.4c
The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification.

ADV_FSP.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_FSP.1.2e
The evaluator shall determine that the functional specification is an accurate and complete instantiation of the SFRs.

5.2.2 Guidance documents (AGD)

5.2.2.1 Operational User Guidance (AGD_OPE.1)

AGD_OPE.1.1d
The developer shall provide operational user guidance.

AGD_OPE.1.1c
The operational user guidance shall describe, for each user role, the user accessible functions and privileges that should be controlled in a secure processing environment, including appropriate warnings.

AGD_OPE.1.2c
The operational user guidance shall describe, for each user role, how to use the available interfaces provided by the TOE in a secure manner.

AGD_OPE.1.3c
The operational user guidance shall describe, for each user role, the available functions and interfaces, in particular all security parameters under the control of the user, indicating secure values as appropriate.

AGD_OPE.1.4c
The operational user guidance shall, for each user role, clearly present each type of security-relevant event relative to the user-accessible functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.

AGD_OPE.1.5c
The operational user guidance shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences, and implications for maintaining secure operation.

AGD_OPE.1.6c
The operational user guidance shall, for each user role, describe the security measures to be followed in order to fulfill the security objectives for the operational environment as described in the ST.

AGD_OPE.1.7c
The operational user guidance shall be clear and reasonable.

AGD_OPE.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.2.2.2 Preparative Procedures (AGD_PRE.1)

AGD_PRE.1.1d
The developer shall provide the TOE, including its preparative procedures.

AGD_PRE.1.1c
The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.

AGD_PRE.1.2c
The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the operational environment in accordance with the security objectives for the operational environment as described in the ST.

AGD_PRE.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AGD_PRE.1.2e
The evaluator shall apply the preparative procedures to confirm that the TOE can be prepared securely for operation.

5.2.3 Life-cycle support (ALC)

5.2.3.1 Labelling of the TOE (ALC_CMC.1)

ALC_CMC.1.1d
The developer shall provide the TOE and a reference for the TOE.

ALC_CMC.1.1c
The TOE shall be labelled with its unique reference.

ALC_CMC.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.2.3.2 TOE CM Coverage (ALC_CMS.1)

ALC_CMS.1.1d
The developer shall provide a configuration list for the TOE.

ALC_CMS.1.1c
The configuration list shall include the following: the TOE itself; and the evaluation evidence required by the SARs.

ALC_CMS.1.2c
The configuration list shall uniquely identify the configuration items.

ALC_CMS.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.2.4 Tests (ATE)

5.2.4.1 Independent Testing - Conformance (ATE_IND.1)

ATE_IND.1.1d
The developer shall provide the TOE for testing.

ATE_IND.1.1c
The TOE shall be suitable for testing.

ATE_IND.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ATE_IND.1.2e
The evaluator shall test a subset of the TSF to confirm that the TSF operates as specified.

5.2.5 Vulnerability assessment (AVA)

5.2.5.1 Vulnerability Survey (AVA_VAN.1)

AVA_VAN.1.1d
The developer shall provide the TOE for testing.

AVA_VAN.1.1c
The TOE shall be suitable for testing.

AVA_VAN.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_VAN.1.2e
The evaluator shall perform a search of public domain sources to identify potential vulnerabilities in the TOE.

AVA_VAN.1.3e
The evaluator shall conduct penetration testing, based on the identified potential vulnerabilities, to determine that the TOE is resistant to attacks performed by an attacker possessing Basic attack potential.

6 TOE Summary Specification

This chapter describes the security functions:

- Security audit
- Cryptographic support
- Identification and authentication
- Security management
- Packet Filtering
- Protection of the TSF
- TOE access
- Trusted path/channels

6.1 Security audit

The TOE is a standalone device that is able to generate and store audit records of security relevant events as they occur. The events that can cause an audit record to be logged include starting and stopping the audit function, any use of an administrator command via the CLI interface, as well as all of the events identified in Table 5-2.

Audit logs are stored as strings and have a format which includes the severity, date and time of the event, the nature or type of the triggering event, an indication of whether the event succeeded, failed or had some other outcome, and the identity of the agent responsible for the event.

The audit records are protected against unauthorized access by only allowing authorized administrators to have access to local audit logs. The logged audit records also include event-specific content that includes at least all of the content required in Table 5-2. For cryptographic keys, the act of importing a key is audited and the associated administrator account that performed the action is recorded.

The TOE supports storage of local audit records which are viewable via the system log via the Web UI. The system log stores up to 150 events after which the audit entries will be overwritten, oldest first.

The TOE components can be configured to use TLS to send audit records to an external syslog server. This transmission happens in real-time.

The TOE includes a hardware clock that is used to provide reliable time information for the audit records it generates.

The TOE audits all IPsec failures including IKE Auth exchange failures due to revoked certificates and authentication failures.

The Security audit function satisfies the following security functional requirements:

- VPNGW13:FAU_GEN.1/VPN, NDcPP22e/MACSEC10:FAU_GEN.1: The TOE generates all the required audit events in Table 5-2 and Table 5-3. Each audit record identifies the date/time, event type, outcome of the event, responsible subject/user, as well as the additional event-specific content indicated in Table 5-2 and Table 5-3.
- NDcPP22e:FAU_GEN.2: The TOE identifies the responsible user for each event based on the specific administrator or network entity (identified by IP address) that caused the event.
- NDcPP22e:FAU_STG.1: Only Authorized Administrators have the ability to view and clear the local system logs.
- NDcPP22e:FAU_STG_EXT.1: The TOE can be configured to export audit records to an external SYSLOG server. This communication is protected with the use of either IPsec or MACsec as determined by the administrator.

6.2 Cryptographic support

A Persistent Systems Wave Relay device is a network appliance with NXP iMX6 Cortex-A9 CPU running software designed to provide the required capabilities. All Wave Relay Devices include the same cryptographic libraries.

- Wave Relay® Kernel Space Crypto Module (HW) version 1.0 (Cert. #A4588)
- Wave Relay® Kernel Space Crypto Module (SW) version 1.0 (Cert. #A4589) All algorithms are supported with PAA and without PAA.
- Wave Relay® User Space Crypto Module version 1.0 (Cert. #A5177). All algorithms are supported with PAA and without PAA

Functions	NIST Standard	Requirement	Certificate #
Encryption/Decryption			
AES GCM (128, 256 bits)	NIST SP 800-38D	FCS_COP.1/DataEncryption	A5177, A4589
AES CBC (128, 256 bits)	NIST SP 800-38A	FCS_COP.1/DataEncryption	A5177, A4588
Cryptographic hashing			
SHA-1, SHA-256, SHA-384, SHA-512	FIPS Pub 180-4	FCS_COP.1/Hash	A5177
Keyed-hash message authentication			
HMAC-SHA-384	FIPS Pub 198-1, "The KeyedHash Message Authentication Code", and FIPS Pub 180-4, "Secure Hash Standard"	FCS_COP.1/KeyedHash	A5177
Cryptographic signature services			
RSA Digital Signature Algorithm (3072 bits)	FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 4	FCS_COP.1/SigGen	A5177
Elliptic Curve Digital Signature Algorithm (P-256, P-384)	[FIPS PUB 186- 4, "Digital Signature Standard (DSS)", Section 5	FCS_COP.1/SigGen	A5177
Random bit generation			
HMAC__DRBG with physical based noise sources with a minimum of 256 bits of non-determinism	HMAC_DRBG	FCS_RBG_EXT.1	A5177
Key generation			
ECC Key Generation (P-256, P-384)	FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4	FCS_CKM.1 FCS_CKM.1/IKE	A5177
Key establishment			
ECC KAS (P-256, P-384)	NIST Special Publication 800-56A, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography"]	FCS_CKM.2	A5177
MACsec			
AES GCM (128 & 256 bits)	NIST SP 800-38D	FCS_COP.1/MACSEC	A4589
AES Key Wrap (128 & 256 bits)	NIST SP 800-38F	FCS_COP.1/MACSEC	A5177
AES CMAC (128 & 256 bits)	NIST SP 800-38B	FCS_COP.1/KeyedHashCMAC	A5177

Table 6-1 TOE Cryptographic Algorithms

The following table outlines key establishment schemes used in the TOE:

Scheme	SFRs	Service

ECC key establishment	FCS_TLSC_EXT.1	Connect to Controlled Network Device
		Syslog
ECC key establishment	FCS_TLSS_EXT.1	Remote Administration
ECC key establishment	FCS_IPSEC_EXT.1	Remote VPN gateways or peers

Table 6-2 Key Establishment Schemes

The TOE supports the following secret keys, private keys and CSPs:

Key	Zeroized upon:	Stored In:	Zeroized By:
TLS host ECDSA private key	Command	Flash	Overwriting once with zeros
TLS host ECDSA digital certificate	Command	Flash	Overwriting once with zeros
TLS pre-master secret	Handshake done	RAM	Overwriting once with zeros
TLS session key	Close of session	RAM	Overwriting once with zeros
MACsec Security Association Key (SAK)	End of session	RAM	Overwriting once with zeros
MACsec Connectivity Association Key (CAK)	Command	Flash	Overwriting once with zeros
MACsec Key Encryption Key (KEK)	End of session	RAM	Overwriting once with zeros
MACsec Integrity Check Key (ICK)	End of session	RAM	Overwriting once with zeros
DH Private Exponent	New key exchange	RAM	Overwritten with new value
User Password	Command	Flash	Overwriting once with zeros
Firmware Integrity / Load RSA public key	Not applicable	Flash	Public value
IPsec Skeyid	End of session	RAM	Overwriting once with zeros
IPsec skeyid_d	End of session	RAM	Overwriting once with zeros
IKE session encrypt key	End of session	RAM	Overwriting once with zeros
IKE session authentication key	End of session	RAM	Overwriting once with zeros
ISAKMP preshared key	Command	Flash	Overwriting once with zeros
IKE ECDSA Private Key	Command	Flash	Overwriting once with zeros
IKE ECDSA digital certificate	Command	Flash	Overwriting once with zeros
IPSec encryption key	End of session	RAM	Overwriting once with zeros
IPSec authentication key	End of session	RAM	Overwriting once with zeros

Table 6-3 Wave Relay Device Keys and CSP

TLS Client Protocol

The TOE includes cryptographic functions that support the TLS v1.2 (RFC 5246) protocol with the following ciphersuites for exporting audit records to a SYSLOG server and for managing an external network device:

- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384

The TOE TLS client implementation offers the following Supported Elliptic Curves/Supported Groups Extensions.

- secp384r1

The TOE can be configured with an X509 certificate which it will send to a TLS server in response to a certificate request message sent by the TLS server. The TOE does not support certificate pinning. The TOE supports the use of IPv4 addresses as reference identifiers within a certificate's Common Name (CN) or Subject Alternate Name (SAN) extension. The TOE checks the SAN/CN when performing certificate validation as described in NDcPP22e:FIA_X509_EXT.1/Rev. Wildcards are not allowed in certificates. IP addresses are converted to binary by parsing decimals delimited by periods. The conversion happens before any comparisons are made. Canonical format is enforced.

TLS Server/HTTPS Protocol

The TOE offers its administrative GUI interface to remote administrators through an HTTPS/TLS connection. The TOE also offers administration from a remote Wave Relay Device peer using API calls on the same TLS port. The TOE implements HTTPS per RFC 2818 with TLSv1.2. A connection can be established only if the remote administrator (peer) initiates the connection.

The following ciphersuites are supported:

- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384

This ciphersuite is not configurable. Key exchanges using secp384r1 are supported. Key exchanges are not configurable. The TOE does not support session resumption.

IPsec Protocol

The TSF implements IPsec as specified in RFCs 3602, 4301, 4303, 4106, 4868, 5996. The TSF supports IKEv2 and ESP connections operating in tunnel mode.

The TSF uses the Linux iptables service to perform IPsec Security Policy Database (SPD) as described in Section 45. Packets not processed by any rules are dropped by a hard-coded final rule that may also log the dropped packet, if configured. Logging of dropped packets is desirable in situations where a network device is expected to filter packets ahead of the VPN. A dropped packet on the VPN gives an indication that the packet filtering configuration may be incorrect.

The TSF can be configured to use the AES-CBC-256 and AES-GCM-256 algorithms for IPsec ESP. When AES-CBC-256 is negotiated as the symmetric cipher, the TOE supports HMAC-SHA-384.

The TSF uses AES-CBC-256 and AES-GCM-256 to encrypt the IKEv2 payload while using HMAC-SHA-384 to authenticate the IKEv2 payload.

The TSF's IPsec configuration does not allow an authorized administrator to select an ESP algorithm with greater strength than the IKE algorithm.

The authorized administrator can configure the TOE to support maximum lifetimes for IKEv2 SAs based on elapsed time. The administrator can specify the maximum lifetime of the Phase 1/IKE SA for 5 minutes, 8 hours, or 24 hours. and specify the maximum lifetime of the Phase 2/ESP SA for 5 minutes, 1 hour, 4 hours, 8 hours, or 24 hours.

The TSF supports DH groups DH Groups 19 and 20 for use with IKEv2. One or more of these groups may be enabled by the administrator. The TSF will negotiate the algorithms in the following order if multiple groups are enabled: DH group 19 and DH group 20. The TSF generates the following ephemeral private key (x) sizes used in Diffie-Hellman based on the negotiated group: 256-bits for DH group 19 and 384-bits for DH group 20.

The TSF negotiates the allowed groups with the client in the IKEv2 exchange. The TSF will not allow the client to use any group not selected in the configuration. For example, if the client has selected group 5, the TSF will refuse to connect because the symmetric strength would be less than 112 bits.

The TSF generates and proposes nonces that are 256 bits long. The nonces are used in the IKEv2 key exchange for all cipher suites and are generated by the DRBG as defined in FCS_RBG_EXT.1. A 256-bit nonce is sufficient to meet the security strengths of all configurable and supported Diffie-Hellman groups, as well as being at least 128 bits and half the strength of the negotiated PRF hash. Because nonces may be created prior to the DH group being chosen, this posture allows the TSF to maximize cryptographic security across all possible key agreement parameters.

The TSF authenticates peers using Pre-shared Keys that conform to RFC 8784. The TSF also supports authenticating peers using ECDSA P-384 X.509 certificates. The TOE will only establish a trusted IPsec channel if the presented identifier in the received certificate matches the configured reference identifier, where the presented and reference identifiers are of the following type: Distinguished Name (DN). Fields within the DN are not individually selectable; the DN must be an exact match for the entire DN string.

MACsec Protocol

The TOE implements MACsec in accordance with IEEE 802.1AE-2006. The TOE derives 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) and rejects any MPDUs that do not contain the identifier. Only EAPOL (PAE EtherType 88-8E) and MACsec frames (EtherType 88-E5) are permitted and others are rejected.

The TOE implements the MACsec requirement for integrity protection with the confidentiality offset of 0. 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. The supported ICV length is 16 octets. An ICV derived with the SAK is used to provide assurance of the integrity of MPDUs. The KDF uses the SCI as the most significant bits of the Initialization Vector (IV) and the 32 least significant bits of the PN as the IV.

The TOE ensures MACsec peer authentication using pre-shared keys. The pre-shared key must be 128 bits when using gcm-aes-128 and 256 bits when using gcm-aes-256. The SAK is generated using the KDF specified in IEEE 802.1X-2010 section 6.2.1. The TOE's random bit generator is used for creating the unique nonces needed for the KDF. The TOE uses AES Key Wrap to distribute the SAKs between peers using aes-128-cmac or aes-256-cmac. The TOE associates Connectivity Association Key Names (CKNs) with CAKs. The length of the CKN shall be an integer number of octets, between 1 and 32 (inclusive). The TOE does not support group CAKs.

The TOE implements Key Agreement Protocol (MKA) in accordance with IEEE 802.1X-2010. The TOE enforces an MKA Lifetime Timeout limit of 6.0 seconds and a MKA Bounded Hello Time limit of 0.5 seconds. Data received in only in Strict-order is accepted by hardware; all others are discarded by the hardware. The TOE verifies the integrity of MKA protocol data units using an ICV derived from the ICK. The ICK is derived from the CAK using KDF (AES-CMAC). The ICV is checked on the reception of each MKA PDU.

The Cryptographic support function satisfies the following security functional requirements:

- NDcPP22e:FCS_CKM.1: The TOE supports asymmetric key generation using ECC key establishment (curves P-256, and P-384), as part of IPsec and TLS as described in the section above.
- VPNGW13:FCS_CKM.1/IKE: See NDcPP22e:FCS_CKM.1
- NDcPP22e:FCS_CKM.2: See Table 6-2 and NDcPP22e:FCS_CKM.1
- NDcPP22e:FCS_CKM.4: All memory is cleared by overwriting it with zeroes.
- NDcPP22e:FCS_COP.1/DataEncryption: The TOE performs encryption and decryption using AES in CBC and GCM mode with key sizes of either 128 or 256.
- VPNGW13:FCS_COP.1/DataEncryption: The TOE performs encryption and decryption using AES in CBC and GCM mode with key sizes of 256 for IPsec.
- MACSEC10:FCS_COP.1/MACSEC: The TOE performs AES key wrap as specified in AES as specified in ISO 18033-3, AES Key Wrap as specified in NIST SP 800-38F, GCM as specified in ISO 19772 with 128 or 256 bit key sizes.
- NDcPP22e:FCS_COP.1/Hash: The TOE supports cryptographic hashing services using SHA-1, SHA-256, SHA-384, and SHA-512 with message digest sizes 160, 256, 384, and 512.
- NDcPP22e:FCS_COP.1/KeyedHash: The TOE supports HMAC-SHA-384 (key and output MAC size 384) for keyed-hash message authentication.
- MACSEC10:FCS_COP.1/KeyedHashCMAC: The TOE supports keyed-hash message authentication in accordance with AES-CMAC algorithm with key sizes 128 bits and 256 bits and the message digest size supported is 128 bits. The algorithm conforms to NIST SP 800-38B.
- NDcPP22e:FCS_COP.1/SigGen: The TOE supports the use of ECDSA with a key size of 384 bits or greater for cryptographic signatures (NIST curve P-384) and RSA 3072 bits.
- NDcPP22e:FCS_HTTPS_EXT.1: The TOE offers its administrative GUI interface to remote administrators through an HTTPS/TLS connection. The TOE implements HTTPS per RFC 2818 with TLSv1.2. A connection can be established only if the remote administrator (peer) initiates the connection.
- NDcPP22e:FCS_IPSEC_EXT.1: The TOE supports IPsec with VPN peers using the IPsec protocol described above.
- VPNGW13:FCS_IPSEC_EXT.1: The TOE supports IPsec when communicating with an external audit server (syslog server) and with VPN peers using the IPsec protocol described above.

- MACSEC10:FCS_MACSEC_EXT.1: The TOE implements MACsec in accordance with IEEE 802.1AE-2006.
- MACSEC10:FCS_MACSEC_EXT.2: The TOE implements the MACsec requirement for integrity protection as described above under the heading of MACsec Protocol.
- MACSEC10:FCS_MACSEC_EXT.3: The TOE generates unique Secure Association Keys (SAKs) using key derivation from Connectivity Association Key (CAK) per section 9.8.1 of IEEE 802.1X-2010 and the TOE's random bit generator as described above.
- MACSEC10:FCS_MACSEC_EXT.4: The TOE ensures MACsec peer authentication as described above under the heading of MACsec Protocol.
- MACSEC10:FCS_MKA_EXT.1: The TOE ensures MACsec peer authentication. See the description above under the heading of MACsec Protocol.
- NDcPP22e:FCS_NTP_EXT.1: The TOE supports NTPv4, authenticating the NTP server that it synchronizes to using SHA1, SHA256, SHA384, SHA512, AES-CBC-128, and AES-CBC-256 as the message digest algorithms. The TOE allows one or more NTP servers to be configured. At least one is required for time synchronization to occur, but more than 3 NTP servers can be specified.
- NDcPP22e:FCS_RBG_EXT.1: The TSF implements a NIST SP 800-90A SHA-512 HMAC_DRBG for generating random bits. The TSF uses its microprocessor hardware noise source (which a kernel driver exposes as /dev/hw_random) to instantiate the Wave Relay Provider SHA-512 HMAC_DRBG (within the OpenSSL library) using a 384-bit seed. That 384-bit seed contains ~316-bits of entropy, more than enough to satisfy the 256-bit strength of the DRBG.
- NDcPP22e:FCS_TLSC_EXT.1: The TOE supports TLS when exporting audit logs to an external server and when managing an external network device. Certificate pining is not supported. The TOE supports IPv4 reference identifiers. Wildcards are not allowed in certificates.
- NDcPP22e:FCS_TLSS_EXT.1: The TOE supports TLS sessions in conjunction with HTTPS for web based administrator access. The TOE also offers administration from a remote Wave Relay Device peer using API calls on the same TLS port.

6.3 Identification and authentication

The TOE provides a password mechanism for authenticating users. Users are associated with a username and password. Users may authenticate remotely via the WebUI or locally to the WMI via the isolated local IP address. Passwords can be composed of any alphabetic, numeric, and a wide range of special characters (identified in FIA_PMG_EXT.1 as follows: '!', '@', '#', '%', '*', '(', ')', '+', ',', '-', '.', ':', ';', ' =', '?', '[', ']', '^', '_', '`', '{', '|', '}', '~'). Passwords are not echoed back when users logon to the TOE. Internally, the TOE keeps track of failed login attempts. If an administrator fails for a configured number of attempts, the administrator is either locked out for a period of time or until the primary administrator unlocks the account. The primary administrator can always log into the device via the local serial CLI connection and can never be locked out from this login.

The TOE requires identification and authentication before allowing access. Only the following functionality is available prior to authentication being completed:

- Display the warning banner in accordance with FTA_TAB.1;
- Display the Node's name and IP address
- Display the Node's battery level
- Display Status via LED
- Provide Rotary knob for Zeroization (Cryptographic Keys, Management Password, HTTPS certificate)
- Change Talkgroups
- Modify Audio Volume

The administrator can configure the number of failed login attempts allowed before the administrator account becomes locked out. The administrator can set a value of 3, 5 or 10 max failure attempts. Once locked out, authentication of the administrator is not allowed to login through the GUI or programmatic interface for the lockout-duration period.

After the lockout-duration period, the use of the GUI and programmatic interface is fully restored. The lockout duration can be set by an administrator to 10 seconds, 1 minute, or 10 minutes.

The TOE verifies x509 certificates that are received from controlled network devices via TLS and from VPN peers via IPsec. Certificates are validated as part of the authentication process for a network peer providing IPSec-based VPN communication (details of the validation are provided immediately below). Certificates presented by a TLS-protected channel to a controlled network device or for secure syslog are also validated as described below. The trust chain of every certificate that is imported into the TOE for use by the TOE is validated when it is first imported into the TOE.

The TOE supports OCSP for X509v3 certificate revocation checking for IPsec and CRL for its TLS client implementation. In addition to a revocation check using OCSP, the following fields are verified as appropriate for TLS and IPsec based peer authentication:

- Expiration – Current time must be within validity period of the certificate.
- Reference Identifier – TLS supports the checking of the IPv4 address in the SAN/CN as described in section 6.2 above. For IPsec, when an incoming request comes in, the TOE matches the configured DN to the peer certificate.
- Key Usage - CA Certificates must have "Certificate Sign", while identity certificates must have "Digital Signature" key usage. CA certificates used to sign CRLs must have the cRLsign key usage bit set.
- X509v3 Extended Key Usage - Must rightly indicate whether Certificate is for use as a "server" certificate or a "client" certificate. If incorrect, connection is not allowed. Certificates used to sign an OCSP response must include the OCSPSigning EKU.
- Authority Information Access (for IPsec) - Must have valid OCSP server respond affirmatively. If this field is not present, an OCSP check is not performed.
- CRL Distribution Point - Must have valid URL to download CRL. If this field is not present, the CRL cannot be retrieved.
- Basics Constraints - Attribute must be present and must have CA flag set to TRUE in all CA certificates.
- Explicit Curves – ECDSA certificates in the chain must use named curves and not curves with explicitly define parameters.

The Identification and authentication function satisfies the following security functional requirements:

- NDcPP22e:FIA_AFL.1: The TOE is capable of locking the administrator account from remote access when the configured threshold of failed login attempts has been exceeded. The account becomes unlocked as described above.
- NDcPP22e:FIA_PMG_EXT.1: The TSF enforces an administrator configurable password length. The minimum password length may be set to 8 to 256 characters. Passwords can be created using any alphabetic, numeric, and the following special characters (!@#%^*()).
- MACSEC10:FIA_PSK_EXT.1: The TOE supports the use of pre-shared keys for MKA as defined by IEEE 802.1X. The pre-shared keys can be generated by the TOE or the TOE will accept bit based pre-shared keys.
- VPNGW13:FIA_PSK_EXT.1: The TOE supports use of IKEv2 pre-shared keys for authentication of IPsec tunnels that conform to RFC 8784. Preshared keys can be entered as bit-based (hex) values. The TOE supports 256 bit PSKs which are 64 hexadecimal characters.
- VPNGW13:FIA_PSK_EXT.2: See FIA_PSK_EXT.1 above
- NDcPP22e:FIA_UAU.7: The TOE does not echo passwords as they are entered; rather ‘*’ characters are echoed when entering passwords.
- NDcPP22e:FIA_UAU_EXT.2: The TOE uses local password-based authentication.

- NDcPP22e:FIA_UIA_EXT.1: The TOE offers limited functions prior to authenticating an administrator as described above. All other administrative actions require the user to be identified and authenticated.
- VPNGW13:FIA_X509_EXT.1/Rev, NDcPP22e:FIA_X509_EXT.1/Rev: The TOE validates x509v3 certificates received during authentication of network peers as described above. Revocation checking is performed using OCSP for IPsec and CRL for TLS.
- VPNGW13:FIA_X509_EXT.2, NDcPP22e:FIA_X509_EXT.2: Revocation checking is performed using OCSP for IPsec and CRL for TLS. If the OCSP responder or CDP identified for a certificate cannot be reached to determine the revocation status of the certificate or the OCSP response/CRL is rejected as invalid, then the certificate is considered not valid and the connection is rejected.
- VPNGW13:FIA_X509_EXT.3, NDcPP22e:FIA_X509_EXT.3: The TOE generates certificate requests and validates the CA used to sign the certificates. Certificates can only be imported if the certificate chains to a known trusted root CA.

6.4 Security management

The TOE supports multiples user associated with a username. All users are administrators with the same privilege level. During a login, the user is required to provide the password.

The TSF does not allow any administrative actions to be performed prior to authentication of the administrative user. Once the administrative user is authenticated, the TSF grants the user access to the administrative GUI. No general-purpose functionality is provided by the TOE. The GUI defines the administrative operations that are available for administering the TSF.

The TSF enforces these restrictions by restricting the administrator to the environment offered by the administrative GUI. When the TSF grants access to an administrative user, the user has full access all operations offered by the GUI.

The TSF restricts the following functions to authorized administrators:

NDcPP22e Required Management Functions
- Ability to administer the TOE locally and remotely,
- Ability to configure the access banner,
- Ability to configure the session inactivity time before session termination or locking,
- Ability to update the TOE, and to verify the updates using digital signature capability prior to installing those updates,
- Ability to configure the authentication failure parameters for FIA_AFL.1,
- Ability to modify the behavior of the transmission of audit data to an external IT entity,
- Ability to manage the cryptographic keys,
- Ability to configure the cryptographic functionality,
- Ability to configure the lifetime for IPsec SAs,
- Ability to set the time which is used for time-stamps,
- Ability to configure NTP,
- Ability to configure the reference identifier for the peer,
- Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors,
- Ability to import X509v3 certificates to the TOE's trust store

VPNGW13 Required Management Functions
- Definition of packet filtering rules,
- Association of packet filtering rules to network interfaces,
- Ordering of packet filtering rules by priority

MACSEC10 Required Management Functions
- Manage a PSK-based CAK and install it in the device;
- Specify a lifetime of a CAK;
- 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);
- Enable, disable, or delete a PSK-based CAK using the MIB object ieee8021XKayMkaPartActivateControl

The Security management function satisfies the following security functional requirements:

- NDcPP22e:FMT_MOF.1/ManualUpdate: Only the authorized administrator can update the TOE by manually initiating the upgrade through the administrative GUI.
- NDcPP22e:FMT_MOF.1/Services: Only the authorized administrator can enable/disable services.
- NDcPP22e:FMT_MTD.1/CoreData: Only the authorized administrator can configure TSF-related functions.
- NDcPP22e:FMT_MTD.1/CryptoKeys: Only the authorized administrator can configure cryptographic keys. The keys an authorized administrator can manage consist of importing trusted Root CA certs, loading X.509 certificates for IPsec and TLS, and PSKs for both MACsec and IPsec. These keys can be also be deleted.
- VPNGW13:FMT_MTD.1/CryptoKeys: see NDcPP22e:FMT_MTD.1/CryptoKeys above
- NDcPP22e:FMT_SMF.1: The TOE allows the administrator to perform the administrative functions identified above.
- VPNGW13:FMT_SMF.1/VPN: The TOE allows the administrator to perform packet filtering related management including definition of rules, ordering of rules, and associating rules with interfaces.
- MACSEC10:FMT_SMF.1/MACSEC: The TOE provides administrative interfaces to perform the functions identified above.
- NDcPP22e:FMT_SMR.2: The TOE maintains administrative user roles.

6.5 Packet Filtering

The TOE supports an Ethernet Interface over which communication for Trusted Channels and Trusted Paths occurs.

During power on, the TOE prevents all network communication prior to completion of the power-up self-tests. This ensures that the TSF is operating properly and that the packet filtering rules have been initialized before the TSF processes any network data. Once the network interfaces have been enabled, packet filtering rules are enforced.

Packet filtering rulesets are integral to the correct functioning of the network packet routing functionality; any failure of the packet filtering component will also cause networking to fail. At any time that networking functionality is enabled, the packet filtering rules will be applied.

The TOE supports the following protocols:

- IPv4 (RFC 791)
- IPv6 (RFC 2460)
- TCP (RFC 793)
- UDP (RFC 768)

The correctness of the TOE's implementation of these protocols is demonstrated through interoperability testing with other devices and network services.

Packets entering the TSF's network stack are filtered by the TOE implementation of iptables. The TOE's iptables examines the following fields within the header of each packet:

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

    - Source Port
    - Destination Port
- UDP (RFC 768)
    - Source Port
    - Destination Port

The IPsec engine then performs its own filtering and processing as necessary to encrypt and decrypt packets. Finally, the resulting packets are processed via iptables once more.

Each packet filtering rule is configured as a permit (pass packets), deny (drop packets silently), reject (drop packets and issue and ICMP response), or log rule. The TOE applies the packet filtering rules in the order they are configured by the administrator. If a packet does not match any of the configured rules, the TOE drops and logs the packet without sending an ICMP response.

The TOE allows packet filter rules to be associated with one or more interfaces. The TOE does not define any interface groups; however, the TOE allows wildcards to be used when specifying the interfaces a rule applies to. FORWARD rules allow packets to be routed from one interface to another.

With the exception of IKE and ESP packets, all received packets destined for the TOE (based on IP address) are first subjected to the INPUT rules associated with the receiving interface. All routed packets are subjected to the FORWARD rules to determine if the packet should be passed on. All packets generated locally by the TOE are subjected to the OUTPUT rules associated with the sending interface.

The Packet Filtering function satisfies the following security functional requirements:

- VPNGW13:FPF_RUL_EXT.1: The TOE supports all of the required protocols, ipv4 (RFC 791), ipv6 (RFC 8200), tcp (RFC 793), and udp (RFC 768) as well as source and destination address. The SPD entries implement permit and deny possibilities. Each SPD entry can be configured to log status of packets pertaining to the entry.

6.6 Protection of the TSF

The TOE does not provide access to locally stored passwords (which can be administratively configured to be protected by SHA-1 or SHA-256) and also, while cryptographic keys can be entered, the TOE does not disclose any cryptographic keys stored in the TOE.

The TSF includes an array of self-tests that are run during startup and periodically during operations to prevent secure data from being released and to ensure all components are functioning correctly. In the event of any self-test failure, the cryptographic library will restart. Self-test Success status is indicated by the status LED as well as via HTTPS. No keys or CSPs will be output when the cryptographic library is in an error state. If the self-tests succeed, the operator will be presented with a login screen when accessing the Wave Relay Device via HTTPS. Attempts to access it via HTTP will be automatically redirected to HTTPS. If the self-tests fail, any attempt to access the cryptographic library will fail.

Self-tests are always run during startup and cannot be disabled. On failure, the cryptographic library will always be non-operational as there is no non-FIPS or bypass mode available. These self-tests verify the integrity of all TSF components by verifying a digital signature (RSA 3072) for each TSF component. These self-tests also include power-on self-tests that verify the correctness of TSF cryptographic operations using known-answer-tests (KAT) which encrypt known data and which verify the result against known good encrypted data. The TOE also performs noise health tests on the DRBG.

The administrator can query the currently running TOE version by viewing the “Node Information” page in the administrative GUI and can perform firmware upgrades using the “Firmware Upgrade” button on the “Node Configuration” tab of the administrative GUI. Firmware can be updated only after the verification of the digital signature within the updated firmware. Persistent Systems signs firmware updates using an RSA 3072 -bit key which the running TOE verifies during an update operation.

The TSF stored public keys used to verify a software update are stored in plaintext on the file system. No interface is provided that allows any user to view, modify or access these public keys except in the context of the verification of

a firmware update. The only method of modifying a public key is to use the trusted update function to update the entire firmware image that contains the key.

The TSF contains a real-time clock. The administrator has the option to either set the time on the clock manually or configure an external NTP server that can be queried periodically to update the clock to ensure that it is accurate and reliable. The TSF utilizes the time for the following security functions:

- Administrative session timeout checking
- Administrative remote authentication lockout timer
- Certificate expiration checking
- Phase 1/IKE SA rekey/expiration interval
- Phase 2/Child SA rekey/expiration interval
- Audit record timestamps
- VPN session inactivity checking
- VPN session establishment checking

The Protection of the TSF function satisfies the following security functional requirements:

- NDcPP22e:FPT_APW_EXT.1: The TOE does not offer any functions that will disclose to any user a plain text password. Furthermore, locally defined passwords are not stored in plaintext form.
- MACSEC10:FPT_CAK_EXT.1: The TOE does not offer any functions that will disclose to any user a CAK.
- MACSEC10:FPT_FLS.1: If the TOE encounters a self-test failure, failure of integrity check of the TSF executable image, failure of noise source health tests it will shut down. The TOE will not restart as long as it has a failure.
- VPNGW13:FPT_FLS.1/SelfTest: If any self-testing generates a failure, the TOE immediately fails-secure by shutting down.
- MACSEC10:FPT_RPL.1: The TOE detects and logs all attempts to replay MPDUs by verifying the packet number (PN) and MKA frames by verifying the message number (MN). If the received PN or MN is lower than the current PN or MN, this indicates to the TOE a replay attempt, and the packet is discarded.
- NDcPP22e:FPT_SKP_EXT.1: The TOE does not offer any functions that will disclose to any users a stored cryptographic key. Keys are stored as identified in Table 6-3 when they are created.
- NDcPP22e:FPT_STM_EXT.1: The TOE includes its own hardware clock which can be set by an administrator and can synchronize its time with an external NTP server.
- NDcPP22e/VPNGW13:FPT_TST_EXT.1: The TOE performs a suite of self-tests to verify its integrity and proper cryptographic functionality.
- VPNGW13:FPT_TST_EXT.3: The TOE performs a suite of self-tests to verify its integrity.
- NDcPP22e/VPNGW13:FPT_TUD_EXT.1: The TOE provides a function to query the current firmware version and upgrade the software embedded in the TOE appliance. When installing updated software, digital signatures are used to authenticate the update to ensure it is the update intended and originated by the vendor. If the digital-signature does not match the expected signature, the administrator should not proceed with the installation. If it matches, the administrator can proceed with the update.

6.7 TOE access

The administrator can access the TSF remotely via the HTTPS/TLS protected GUI and Programmatic interface. The TSF displays a configurable advisory and consent message when an administrator accesses the GUI interface. The administrator can terminate a GUI session by logging out.

The TOE locks local administrative sessions after an administrator configured period of inactivity. The inactivity period can be configured to be 20 seconds, 5 minutes, 10 minutes or 20 minutes (the default). The TSF blanks the local console when it locks the session and requires the administrator to re-authenticate before allowing administrative access.

The TOE terminates remote administrative sessions after an administrator configured period of inactivity. The inactivity period can be configured to be 20 seconds, 5 minutes, 10 minutes or 20 minutes (the default).

The HTTPS/TLS protected Programmatic interface automatically terminates the session once the configuration operations are complete.

The TOE supports a dedicate local IP address for local management.

The TOE access function satisfies the following security functional requirements:

- NDcPP22e:FTA_SSL.3: The TOE terminates remote sessions that have been inactive for an administrator-configured period of time.
- VPNGW13:FTA_SSL.3.1/VPN: The TOE monitors the client VPN activity within the Phase 2/Child SA. If no packets are sent for an administrator configured period of time, the TOE terminates the client VPN.
- NDcPP22e:FTA_SSL.4: The TOE provides the function to logout (or terminate) both local and remote user sessions as directed by the user. The admin must press the “logout” button located on the top right of the Web UI to terminate the session.
- NDcPP22e:FTA_SSL_EXT.1: The TOE terminates local sessions that have been inactive for an administrator-configured period of time.
- NDcPP22e:FTA_TAB.1: The TOE can be configured to display administrator-defined advisory banners when administrators successfully establish interactive sessions with the TOE, allowing unauthorized users to terminate their session prior to performing any functions.

6.8 Trusted path/channels

The TOE offers an HTTPS/TLS protected GUI interface to remote administrative users which can be used to configure the TOE. The TOE also offers administration from a remote Wave Relay Device peer using API calls on the same TLS port. When a remote administrator attempts to connect to the TOE using either the GUI or Wave Relay the TOE attempts to negotiate a TLS session. If the session cannot be negotiated, the connection is dropped.

The TOE can be configured to establish MACsec connections with MACsec capable peers thereby ensuring that communication with a MACsec peer is protected from disclosure and modification.

The TOE protects communication to an external audit server (syslog server) using TLS. The TOE can also communicate with a controlled network device (which is another Wave Relay Device) using the same TLS port. The TLS protocol prevents disclosure and modification of communication with the controlled network device.

Finally, the TOE can provide IPsec protected communications to VPN peers thereby ensuring that communication with a VPN peer is protected from disclosure and modification.

Service	TOE Role	Protocol	Type
Audit Server	Initiator	TLS	Trusted Channel
MACsec peer	Initiator or Responder	MACsec	Trusted Channel
Controlled Network Device	Client	TLS	Trusted Channel
Remote VPN gateways or peers	Initiator or Responder	IPsec	Trusted Channel
Remote Administrator GUI/ Remote Administrator Programmatic APIs	Server	TLS/HTTPS	Trusted Path

Table 6-4 Trusted Channels and Trusted Paths

The Trusted path/channels function satisfies the following security functional requirements:

- NDcPP22e:FTP_ITC.1: The TOE uses IPsec when exporting audit records to an external audit server (syslog server). The TOE also uses IPsec when communicating with IPsec clients and peers.
- VPNGW13:FTP_ITC.1/VPN: When making connections with remote VPN clients and peers the TOE initiates the IPsec communication between the peers.
- MACSEC10:FTP_ITC.1: In the evaluated configuration, the TOE can be configured to use IPsec or MACsec to ensure that any exported audit records and authentication server communications are protected so they are not subject to inappropriate disclosure or modification. The TOE can be configured to establish MACsec connections with MACsec capable peers.
- NDcPP22e:FTP_TRP.1/Admin: The TOE offers an HTTPS/TLS protected GUI interface to remote administrative users which can be used to configure the TOE. The TOE can also be accessed from another Wave Relay Device on the TLS port. The TLS protocol protects communication from disclosure and modification.
- MACSEC10:FTP_TRP.1: See NDcPP22e:FTP_TRP.1/Admin.