mTera Universal Transport Platform version MT5.1.2 Security Target

Version 0.5
08/26/2021

Prepared for:

Infinera Corporation

9005 Junction Dr, Suite C
Annapolis Junction, MD 20701

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 mTera Universal Transport Platform provided by Infinera Corporation. The TOE is being evaluated as a Network 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 – mTera Universal Transport Platform version MT5.1.2 Security Target

ST Version – Version 0.5

ST Date – 08/26/2021

1.2 TOE Reference

TOE Identification – Infinera Corporation mTera Universal Transport Platform

The TOE includes the following hardware:

Product Model	Part Number(s)	CPU
Infinera Corporation mTera Universal Transport Platform – 8 slot chassis	81.71S-MTERA8-R6	See the 8-slot STPM CPU below

Product Model	Part Number(s)	CPU
Infinera Corporation mTera Universal Transport Platform – 16 slot chassis	81.71S-MTERA-R6	See the 16-slot STPM CPU below
8-slot STPM	82.71C-M8STPM-R6	NXP QorIQ P-4080
16-slot STPM	82.71C-MSTPM-R6	NXP QorIQ P-4080

The STPM cards are management cards that run the software image for the mTera. The only difference between the two STPMs is that they are built with different form factors.

TOE Developer – Infinera Corporation

Evaluation Sponsor – Infinera Corporation

1.3 TOE Overview

The Target of Evaluation (TOE) is the Infinera mTera version MT5.1.2. The mTera is an optical network appliance delivering Wavelength, High-capacity Electrical OTN, and Packet network switching. The mTera supports electrical switching using an agnostic switch fabric. Signals switched by the electrical switch fabric include high-capacity ITU Optical Transport Network (OTN) ODU switching, ITU/ANSI SDH/SONET switching and service oriented MPLS-TP/Ethernet packet switching. The security functions provided include Identification, Authentication, Access Control, Protection of TSF, Confidentiality, Integrity and Auditing.

The Infinera mTera has two shelf form factors: the 16-slot mTera and the 8-slot mTera8. The same software image is used on both systems, delivering consistent security functionality across a network of mTera systems. Both mTeras use the same management cards, and the only difference is the number of slots.

The Infinera mTera has 3 classes of ports for security management: the Local Craft Interface (1000bT Ethernet), the DCN Interface (1000bT Ethernet) and in-band management channels (OTN OSC, OTN GCC, SDH/SONET DCC, Management VLAN). The functionality of in-band management channels is the same as the DCN interface, except these channels are part of an optical networking switch logic that allows access to the management interface of the management card. The Local Craft Interface is a non-routable network limited to interaction with a local workstation and provides a trusted interface for local operations. The DCN Interface is a routed interface used to attach to a service provider's management network. The in-band management channels are routed interfaces used to interconnect other optical networking systems. The Infinera mTera provides configuration access, time-of-day synchronization, and audit reporting over these interfaces.

1.4 TOE Description

The Infinera mTera Universal Transport Platform is an extremely flexible and highly efficient transport solution supporting up to 12Tb/s of switching and grooming for OTN, Packet and SONET/SDH leveraging protocol agnostic fabrics and interface cards that can be software configured for OTN, MPLS-TP or Carrier Ethernet on each interface or virtual interface. The Infinera mTera is offered in either a 16-slot chassis or 8-slot chassis.

1.4.1 TOE Architecture

The Infinera mTera is a network appliance composed of various module slots. The mTera supports up to two STPM (shelf timing and processor module) cards for management in addition to optical network switch blades. The mTera TOE's management capabilities rely on TL1 commands. The mTera's interfaces provide TLS and IPsec functionality as well as SSHv2 for management.

The mTera STPM includes an LCI (local craft interface) port for local CLI access and DCN (data communications network) ports for network communications. Though an administrator uses SSH to connect to the mTera through both interfaces, the LCI port does not provide network routing, thus requiring an administrator to connect directly to the LCI port for local console access.

The mTera TOE contains an OpenSSL FIPS object module and a kernel crypto module for cryptographic services. The Kernel Crypto module is used for IPSEC datapath (encrypt/decrypt, message authentication, hash) while the OpenSSL module is used for IPSEC Key management (i.e. IKE), SSH and TLS operations. The OpenSSL module is used for secret negotiation, authentication, encrypt/decrypt, message authentication, hashes and drbg.

1.4.1.1 Physical Boundaries

The TOE's physical boundary includes the entire chassis, which contains the STPM management and optical network line cards. The TOE runs firmware version FP.5.1.2p1.

The TOE operates with the following components in the Operating Environment:

- Audit Server – The TOE utilizes an external syslog server to store audit records.
- Authentication Server – The TOE has the ability to use RADIUS servers to authenticate users.
- Time Server – The TOE uses a Network Time Protocol (NTP) server to synchronize its system clock with a central time source.
- SSH Client – The remote administrator uses an SSH client to access the CLI.

1.4.1.2 Logical Boundaries

This section summarizes the security functions provided by mTera:

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

1.4.1.2.1 Security audit

The TOE is designed to be able to generate logs for a wide range of security relevant events including start-up and shutdown of the TOE, all administrator actions, and all events identified in Table 2 Auditable Events. The TOE can be configured to store the logs locally so they can be accessed by an administrator or alternately to send the logs to a designated syslog server in the operational environment.

1.4.1.2.2 Cryptographic support

The TOE includes cryptographic modules that provide key management, random bit generation, encryption/decryption, digital signature and secure hashing and key-hashing features in support of higher level cryptographic protocols including IPsec, SSH, and TLS.

1.4.1.2.3 Identification and authentication

The TOE requires administrators to be identified and authenticated before they can access any TOE security functions. The TOE supports role-based authentication, so user accounts are assigned predefined roles which restrict them based on their assigned role. The TOE maintains these administrator and user attributes which can be defined locally with user names and passwords or can be defined in the context of local RADIUS services. Authentication can be either locally or remotely through an external authentication server, or internally. After an administrator-specified number of failed attempts, the user account is locked out. The TOE's password mechanism provides configuration for a minimum password length. The TOE also protects, stores and allows authorized administrators to load X.509.v3 certificates for use to support authentication for IPsec and TLS connections.

1.4.1.2.4 Security management

The TOE provides the administrator role the capability to configure and manage all TOE security functions including cryptographic operations, user accounts, passwords, advisory banner, session inactivity and TOE updates. The management functions are restricted to the administrator role. The role must have the appropriate access privileges or access will be denied. The TOE's cryptographic functions ensure that only secure values are accepted for security attributes.

1.4.1.2.5 Protection of the TSF

The TOE has its own internal hardware clock that provides reliable time stamps used for auditing. The TOE stores passwords on flash and encrypts the passwords using an AES-256-CBC key. The TOE does not provide any interfaces that allow passwords or keys to be read. The TOE also provides integrity and security protection for all communication between its components. This prevents unauthorized modification or disclosure of TSF data during transmission.

The TOE runs self-tests during power up and periodically during operation to ensure the correct operation of the cryptographic functions and TSF hardware. There is an option for the administrator to verify the integrity of stored TSF executable code. The TOE executes self-tests for both the Kernel Crypto module and OpenSSL FIPS Object module.

The TOE includes mechanisms so that the administrator can determine the TOE version and update the TOE securely using digital signatures.

1.4.1.2.6 TOE access

The TOE allows administrators to configure a period of inactivity for administrator and user sessions. Once that time period has been reached while the session has no activity, the session is terminated. All users may also terminate their own sessions at any time. A warning banner is displayed at the management interfaces (local CLI and SSH) to advise users on appropriate use and penalty for misuse of system.

1.4.1.2.7 Trusted path/channels

The TOE uses IPsec to provide an encrypted channel between itself and third-party trusted IT entities in the operating environment including external syslog server, external authentication server and NTP server. The TOE uses TLS to secure network communications with an external optical network peer.

The TOE secures remote communication with administrators by implementing SSHv2 for CLI access. Both the integrity and disclosure protection are ensured via the secure protocol. If the negotiation of a secure session fails or if the user cannot be authenticated for remote administration, the attempted session will not be established.

1.4.2 TOE Documentation

The TOE includes the following guidance documents:

- Coriant Product Hardening Guide, version BP11, August 23, 2021
- TL1 Specification, version CP11, August 26, 2021

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:
    - collaborative Protection Profile for Network Devices, Version 2.1, 24 September 2018 (NDcPP21)
- NIAP Technical Decisions:

Technical Decision	Applied?
0572 – NiT Technical Decision for Restricting FTP_ITC.1 to only IP address identifiers	Yes
0571 – NiT Technical Decision for Guidance on how to handle FIA_AFL.1	Yes
0570 – NiT Technical Decision for Clarification about FIA_AFL.1	Yes
0547 – NIT Technical Decision for Clarification on developer disclosure of AVA_VAN	Yes
0538 – NIT Technical Decision for Outdated link to allowed-with list	Yes
0536 – NIT Technical Decision for Update Verification Inconsistency	Yes
0535 – NIT Technical Decision for Clarification about digital signature algorithms for FTP_TUD.1	Yes
0533 – NIT Technical Decision for FTP_ITC.1 with signed downloads	Yes
0532 – NIT Technical Decision for Use of seeds with higher entropy	Yes
0531 – NIT Technical Decision for Challenge-Response for Authentication	Yes
0530 – NIT Technical Decision for FCS_TLSC_EXT.1.1 5e test clarification	No
0529 – NIT Technical Decision for OCSP and Authority Information Access extension	No
0528 – NIT Technical Decision for Missing EAs for FCS_NTP_EXT.1.4	Yes
0484 – NIT Technical Decision for Interactive sessions in FTA_SSL_EXT.1 & FTA_SSL.3	Yes
0483 – NIT Technical Decision for Applicability of FPT_APW_EXT.1	Yes
0482 – NIT Technical Decision for Identification of usage of cryptographic schemes	Yes
0481 – NIT Technical Decision for FCS_(D)TLSC_EXT.X.2 IP addresses in reference identifiers	Yes
0480 – NIT Technical Decision for Granularity of audit events	Yes
0478 – NIT Technical Decision for Application Notes for FIA_X509_EXT.1 iterations	Yes
0477 – NIT Technical Decision for Clarifying FPT_TUD_EXT.1 Trusted Update	Yes
0475 – NIT Technical Decision for Separate traffic consideration for SSH rekey	Yes
0453 – NIT Technical Decision for Clarify authentication methods SSH clients can use to authenticate SSH se	No
0451 – NIT Technical Decision for ITT Comm UUID Reference Identifier	Yes
0450 – NIT Technical Decision for RSA-based ciphers and the Server Key Exchange message	Yes
0447 – NIT Technical Decision for Using 'diffie-hellman-group-exchange-sha256' in FCS_SSHC/S_EXT.1.7	Yes
0425 – NIT Technical Decision for Cut-and-paste Error for Guidance AA	Yes
0424 – NIT Technical Decision for NDcPP v2.1 Clarification - FCS_SSHC/S_EXT1.5	Yes

0423 – NIT Technical Decision for Clarification about application of Rfl#201726rev2	Yes
0412 – NIT Technical Decision for FCS_SSHS_EXT.1.5 SFR and AA discrepancy	Yes
0411 – NIT Technical Decision for FCS_SSHC_EXT.1.5, Test 1 - Server and client side seem to be confused	No
0410 – NIT technical decision for Redundant assurance activities associated with FAU_GEN.1	Yes
0409 – NIT decision for Applicability of FIA_AFL.1 to key-based SSH authentication	Yes
0408 – NIT Technical Decision for local vs. remote administrator accounts	Yes
0407 – NIT Technical Decision for handling Certification of Cloud Deployments	No
0402 – NIT Technical Decision for RSA-based FCS_CKM.2 Selection	Yes
0401 – NIT Technical Decision for Reliance on external servers to meet SFRs	Yes
0400 – NIT Technical Decision for FCS_CKM.2 and elliptic curve-based key establishment	Yes
0399 – NIT Technical Decision for Manual installation of CRL (FIA_X509_EXT.2)	Yes
0398 – NIT Technical Decision for FCS_SSH*EXT.1.1 RFCs for AES-CTR	Yes
0397 – NIT Technical Decision for Fixing AES-CTR Mode Tests	Yes
0396 – NIT Technical Decision for FCS_TLSC_EXT.1.1, Test 2	Yes
0395 – NIT Technical Decision for Different Handling of TLS1.1 and TLS1.2	Yes

2.1 Conformance Rationale

The ST conforms to the NDcPP21. 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 NDcPP21 and this section reproduces only the corresponding Security Objectives for operational environment for reader convenience. The NDcPP21 offers additional information about the identified security objectives, but that has not been reproduced here and the NDcPP21 should be consulted if there is interest in that material.

In general, the NDcPP21 has defined Security Objectives appropriate for Network Devices and as such are applicable to the mTera Universal Transport Platform 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.NO_GENERAL_PURPOSE There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE.

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

OE.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.

OE.TRUSTED_ADMIN TOE Administrators are trusted to follow and apply all guidance documentation in a trusted manner.
For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are assumed to monitor the revocation status of all certificates in the TOE's trust store and to remove any certificate from the TOE's trust store in case such certificate can no longer be trusted.

OE.UPDATES The TOE firmware and software is updated by an administrator on a regular basis in response to the release of product updates due to known vulnerabilities.

4. Extended Components Definition

All of the extended requirements in this ST have been drawn from the NDcPP21. The NDcPP21 defines the following extended requirements and since they are not redefined in this ST the NDcPP21 should be consulted for more information in regard to those CC extensions.

Extended SFRs:

- NDcPP21:FAU_STG_EXT.1: Protected Audit Event Storage
- NDcPP21:FCS_IPSEC_EXT.1: IPsec Protocol
- NDcPP21:FCS_NTP_EXT.1: NTP Protocol
- NDcPP21:FCS_RBG_EXT.1: Random Bit Generation
- NDcPP21:FCS_SSHS_EXT.1: SSH Server Protocol
- NDcPP21:FCS_TLSC_EXT.2: TLS Client Protocol with authentication
- NDcPP21:FCS_TLSS_EXT.2: TLS Server Protocol with mutual authentication
- NDcPP21:FIA_PMG_EXT.1: Password Management
- NDcPP21:FIA_UAU_EXT.2: Password-based Authentication Mechanism
- NDcPP21:FIA_UIA_EXT.1: User Identification and Authentication
- NDcPP21:FIA_X509_EXT.1/Rev: X.509 Certificate Validation
- NDcPP21:FIA_X509_EXT.2: X.509 Certificate Authentication
- NDcPP21:FIA_X509_EXT.3: X.509 Certificate Requests
- NDcPP21:FPT_APW_EXT.1: Protection of Administrator Passwords
- NDcPP21:FPT_SKP_EXT.1: Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)
- NDcPP21:FPT_STM_EXT.1: Reliable Time Stamps
- NDcPP21:FPT_TST_EXT.1: TSF testing
- NDcPP21:FPT_TUD_EXT.1: Trusted update
- NDcPP21:FTA_SSL_EXT.1: TSF-initiated Session Locking

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 NDcPP21. The refinements and operations already performed in the NDcPP21 are not identified (e.g., highlighted) here, rather the requirements have been copied from the NDcPP21 and any residual operations have been completed herein. Of particular note, the NDcPP21 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 NDcPP21 which includes all the SARs for EAL 1. However, the SARs are effectively refined since requirement-specific 'Assurance Activities' are defined in the NDcPP21 that serve to ensure corresponding evaluations will yield more practical and consistent assurance than the EAL 1 assurance requirements alone. The NDcPP21 should be consulted for the assurance activity definitions.

5.1 TOE Security Functional Requirements

The following table identifies the SFRs that are satisfied by mTera Universal Transport Platform TOE.

Requirement Class	Requirement Component
FAU: Security audit	NDcPP21:FAU_GEN.1: Audit Data Generation
	NDcPP21:FAU_GEN.2: User identity association
	NDcPP21:FAU_STG_EXT.1: Protected Audit Event Storage
FCS: Cryptographic support	NDcPP21:FCS_CKM.1: Cryptographic Key Generation
	NDcPP21:FCS_CKM.2: Cryptographic Key Establishment
	NDcPP21:FCS_CKM.4: Cryptographic Key Destruction
	NDcPP21:FCS_COP.1/DataEncryption: Cryptographic Operation (AES Data Encryption/Decryption)
	NDcPP21:FCS_COP.1/Hash: Cryptographic Operation (Hash Algorithm)
	NDcPP21:FCS_COP.1/KeyedHash: Cryptographic Operation (Keyed Hash Algorithm)
	NDcPP21:FCS_COP.1/SigGen: Cryptographic Operation (Signature Generation and Verification)
	NDcPP21:FCS_IPSEC_EXT.1: IPsec Protocol
	NDcPP21:FCS_NTP_EXT.1: NTP Protocol
	NDcPP21:FCS_RBG_EXT.1: Random Bit Generation
	NDcPP21:FCS_SSHS_EXT.1: SSH Server Protocol
	NDcPP21:FCS_TLSC_EXT.2: TLS Client Protocol with authentication
	NDcPP21:FCS_TLSS_EXT.2: TLS Server Protocol with mutual authentication
FIA: Identification and authentication	NDcPP21:FIA_AFL.1: Authentication Failure Management
	NDcPP21:FIA_PMG_EXT.1: Password Management
	NDcPP21:FIA_UAU.7: Protected Authentication Feedback
	NDcPP21:FIA_UAU_EXT.2: Password-based Authentication Mechanism
	NDcPP21:FIA_UIA_EXT.1: User Identification and Authentication
	NDcPP21:FIA_X509_EXT.1/Rev: X.509 Certificate Validation
	NDcPP21:FIA_X509_EXT.2: X.509 Certificate Authentication
	NDcPP21:FIA_X509_EXT.3: X.509 Certificate Requests

FMT: Security management	NDcPP21:FMT_MOF.1/ManualUpdate: Management of security functions behaviour
	NDcPP21: FMT_MTD.1/CryptoKeys: Management of TSF Data
	NDcPP21:FMT_MTD.1/CoreData: Management of TSF Data
	NDcPP21: FMT_MOF.1/Functions: Management of security functions behavior
	NDcPP21: FMT_MOF.1/Services: Management of security functions behaviour
	NDcPP21:FMT_SMF.1: Specification of Management Functions
	NDcPP21:FMT_SMR.2: Restrictions on Security Roles
FPT: Protection of the TSF	NDcPP21:FPT_APW_EXT.1: Protection of Administrator Passwords
	NDcPP21:FPT_SKP_EXT.1: Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)
	NDcPP21:FPT_STM_EXT.1: Reliable Time Stamps
	NDcPP21:FPT_TST_EXT.1: TSF testing
	NDcPP21:FPT_TUD_EXT.1: Trusted update
FTA: TOE access	NDcPP21:FTA_SSL.3: TSF-initiated Termination
	NDcPP21:FTA_SSL.4: User-initiated Termination
	NDcPP21:FTA_SSL_EXT.1: TSF-initiated Session Locking
	NDcPP21:FTA_TAB.1: Default TOE Access Banners
FTP: Trusted path/channels	NDcPP21:FTP_ITC.1: Inter-TSF trusted channel
	NDcPP21:FTP_TRP.1/Admin: Trusted Path

Table 1 TOE Security Functional Components

5.1.1 Security audit (FAU)

5.1.1.1 Audit Data Generation (NDcPP21:FAU_GEN.1)

NDcPP21: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 2.

Requirement	Auditable Events	Additional Content
NDcPP21:FAU_GEN.1	None	None
NDcPP21:FAU_GEN.2	None	None
NDcPP21:FAU_STG_EXT.1	None	None
NDcPP21:FCS_CKM.1	None	None
NDcPP21:FCS_CKM.2	None	None
NDcPP21:FCS_CKM.4	None	None

NDcPP21:FCS_COP.1/DataEncryption	None	None
NDcPP21:FCS_COP.1/Hash	None	None
NDcPP21:FCS_COP.1/KeyedHash	None	None
NDcPP21:FCS_COP.1/SigGen	None	None
NDcPP21:FCS_IPSEC_EXT.1	Failure to establish an IPsec SA.	Reason for failure.
NDcPP21:FCS_NTP_EXT.1	Configuration of a new time server Removal of configured time server	Identity if new/removed time server
NDcPP21:FCS_RBG_EXT.1	None	None
NDcPP21:FCS_SSHS_EXT.1	Failure to establish an SSH session.	Reason for failure.
NDcPP21:FCS_TLSC_EXT.2	Failure to establish a TLS Session.	Reason for failure.
NDcPP21:FCS_TLSS_EXT.2	Failure to establish a TLS Session.	Reason for failure.
NDcPP21:FIA_AFL.1	Unsuccessful login attempt limit is met or exceeded.	Origin of the attempt (e.g., IP address).
NDcPP21:FIA_PMG_EXT.1	None	None
NDcPP21:FIA_UAU.7	None	None
NDcPP21:FIA_UAU_EXT.2	All use of identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
NDcPP21:FIA_UIA_EXT.1	All use of identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
NDcPP21: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
NDcPP21:FIA_X509_EXT.2	None	None
NDcPP21:FIA_X509_EXT.3	None	None
NDcPP21:FMT_MOF.1/ManualUpdate	Any attempt to initiate a manual update.	None
NDcPP21:FMT_MTD.1/CryptoKeys	None	None
NDcPP21:FMT_MTD.1/CoreData	None	None
NDcPP21:FMT_MTD.1/Functions	None	None
NDcPP21:FMT_MTD.1/Services	None	None
NDcPP21:FMT_SMF.1	All management activities of TSF data.	None
NDcPP21:FMT_SMR.2	None	None
NDcPP21:FPT_APW_EXT.1	None	None
NDcPP21:FPT_SKP_EXT.1	None	None
NDcPP21: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).
NDcPP21:FPT_TST_EXT.1	None	None
NDcPP21:FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure).	

NDcPP21:FTA_SSL.3	The termination of a remote session by the session locking mechanism.	None
NDcPP21:FTA_SSL.4	The termination of an interactive session.	None
NDcPP21: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
NDcPP21:FTA_TAB.1	None	None
NDcPP21: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.
NDcPP21:FTP_TRP.1/Admin	Initiation of the trusted path. Termination of the trusted path. Failure of the trusted path functions.	None

Table 2 Audit Events

NDcPP21: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 2.

5.1.1.2 User identity association (NDcPP21:FAU_GEN.2)

NDcPP21: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.

5.1.1.3 Protected Audit Event Storage (NDcPP21:FAU_STG_EXT.1)

NDcPP21: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.

NDcPP21:FAU_STG_EXT.1.2

The TSF shall be able to store generated audit data on the TOE itself.

[***TOE shall consist of a single standalone component that stores audit data locally,***]

NDcPP21:FAU_STG_EXT.1.3

The TSF shall [***overwrite previous audit records according to the following rule: [overwrite the earliest 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 (NDcPP21:FCS_CKM.1)

NDcPP21:FCS_CKM.1.1

The TSF shall generate asymmetric cryptographic keys in accordance with a specified cryptographic key generation algorithm: [***- RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, 'Digital Signature Standard (DSS)',***

Appendix B.3,
- ECC schemes using 'NIST curves' [P-256, P-384, P-521] that meet the following: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Appendix B.4,
- FFC Schemes using Diffie-Hellman group 14 that meet the following: RFC 3526, Section 3].

5.1.2.2 Cryptographic Key Establishment (NDcPP21:FCS_CKM.2)

NDcPP21: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 2, 'Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography',***
-Key establishment scheme using Diffie-Hellman group 14 that meets the following: RFC 3526, Section 3]. (TD0402 applied)

5.1.2.3 Cryptographic Key Destruction (NDcPP21:FCS_CKM.4)

NDcPP21: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 [a pseudo-random pattern using the TSF's RBG]***];
- 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 [***o logically addresses the storage location of the key and performs a [single] overwrite consisting of [a pseudo-random pattern using the TSF's RBG]***]

that meets the following: No Standard.

5.1.2.4 Cryptographic Operation (AES Data Encryption/Decryption) (NDcPP21:FCS_COP.1/DataEncryption)

NDcPP21:FCS_COP.1.1/DataEncryption

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

5.1.2.5 Cryptographic Operation (Hash Algorithm) (NDcPP21:FCS_COP.1/Hash)

NDcPP21: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***] that meet the following: ISO/IEC 10118-3:2004.

5.1.2.6 Cryptographic Operation (Keyed Hash Algorithm) (NDcPP21:FCS_COP.1/KeyedHash)

NDcPP21:FCS_COP.1.1/KeyedHash

The TSF shall perform keyed-hash message authentication in accordance with a specified cryptographic algorithm [***HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512***] and cryptographic key sizes [***160 bits for HMAC-SHA-1, 256 bits for HMAC-SHA-256, 384 bits for HMAC-SHA-384 and 512 bits for HMAC-SHA-512***] and message digest sizes [***160, 256, 384, 512***] bits that meet the following: ISO/IEC 9797-2:2011, Section 7 'MAC Algorithm 2'.

5.1.2.7 Cryptographic Operation (Signature Generation and Verification) (NDcPP21:FCS_COP.1/SigGen)

NDcPP21:FCS_COP.1.1/SigGen

The TSF shall perform cryptographic signature services (generation and verification) in

accordance with a specified cryptographic algorithm [***- RSA Digital Signature Algorithm and cryptographic key sizes (modulus) [2048 bits],***
- Elliptic Curve Digital Signature Algorithm and cryptographic key sizes [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-384]; ISO/IEC 14888-3, Section 6.4].

5.1.2.8 IPsec Protocol (NDcPP21:FCS_IPSEC_EXT.1)

NDcPP21:FCS_IPSEC_EXT.1.1
The TSF shall implement the IPsec architecture as specified in RFC 4301.

NDcPP21: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.

NDcPP21:FCS_IPSEC_EXT.1.3
The TSF shall implement [***transport mode, tunnel mode***].

NDcPP21:FCS_IPSEC_EXT.1.4
The TSF shall implement the IPsec protocol ESP as defined by RFC 4303 using the cryptographic algorithms [***AES-CBC-128, AES-CBC-256 (specified by RFC 3602)***] together with a Secure Hash Algorithm (SHA)-based HMAC [***HMAC-SHA-1, HMAC-SHA-384***] and [***AES-GCM-256 (specified in RFC 4106)***].

NDcPP21:FCS_IPSEC_EXT.1.5
The TSF shall implement the protocol: [***- IKEv2 as defined in RFC 5996 and [with no support for NAT traversal], and [RFC 4868 for hash functions]***].

NDcPP21:FCS_IPSEC_EXT.1.6
The TSF shall ensure the encrypted payload in the [***IKEv2***] protocol uses the cryptographic algorithms [***AES-CBC-128, AES-CBC-256 (specified in RFC 3602), AES-GCM-256 (specified in RFC 5282)***].

NDcPP21:FCS_IPSEC_EXT.1.7
The TSF shall ensure that [***- IKEv2 SA lifetimes can be configured by a Security Administrator based on [o length of time, where the time values can be configured within [up to 24 hours] hours]***].

NDcPP21:FCS_IPSEC_EXT.1.8
The TSF shall ensure that [***- IKEv2 Child SA lifetimes can be configured by a Security Administrator based on [o number of bytes,***
o length of time, where the time values can be configured within [up to 24 hours] hours]].

NDcPP21: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 bits***] bits.

NDcPP21:FCS_IPSEC_EXT.1.10
The TSF shall generate nonces used in [***IKEv2***] exchanges of length [***- at least 128 bits in size and at least half the output size of the negotiated pseudorandom function (PRF) hash***].

NDcPP21:FCS_IPSEC_EXT.1.11
The TSF shall ensure that all IKE protocols implement DH Group(s) [***14 (2048-bit MODP), 20 (384-bit Random ECP)***].

NDcPP21: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.

NDcPP21:FCS_IPSEC_EXT.1.13
The TSF shall ensure that all IKE protocols perform peer authentication using [***RSA, ECDSA***] that use X.509v3 certificates that conform to RFC 4945 and [***no other method***].

NDcPP21: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***].

5.1.2.9 NTP Protocol (NDcPP21:FCS_NTP_EXT.1)

NDcPP21:FCS_NTP_EXT.1.1
The TSF shall use only the following NTP version(s) [***NTP v4 (RFC 5905)***].

NDcPP21:FCS_NTP_EXT.1.2
The TSF shall update its system time using [***[IPsec] to provide trusted communication between itself and an NTP time source.***].

NDcPP21:FCS_NTP_EXT.1.3
The TSF shall not update NTP timestamp from broadcast and/or multicast addresses

NDcPP21:FCS_NTP_EXT.1.4
The TSF shall support configuration of at least three (3) NTP time sources.

5.1.2.10 Random Bit Generation (NDcPP21:FCS_RBG_EXT.1)

NDcPP21:FCS_RBG_EXT.1.1
The TSF shall perform all deterministic random bit generation services in accordance with ISO/IEC 18031:2011 using [***CTR_DRBG (AES)***].

NDcPP21:FCS_RBG_EXT.1.2
The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [***[one] hardware-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.

5.1.2.11 SSH Server Protocol (NDcPP21:FCS_SSHS_EXT.1)

NDcPP21:FCS_SSHS_EXT.1.1
The TSF shall implement the SSH protocol that complies with RFC(s) [***4251, 4252, 4253, 4254, 4344, 5656, 6668, 8332***]. (TD0398 applied)

NDcPP21:FCS_SSHS_EXT.1.2
The TSF shall ensure that the SSH protocol implementation supports the following authentication methods as described in RFC 4252: public key-based, [***password-based***].

NDcPP21:FCS_SSHS_EXT.1.3
The TSF shall ensure that, as described in RFC 4253, packets greater than [***262144***] bytes in an SSH transport connection are dropped.

NDcPP21:FCS_SSHS_EXT.1.4
The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms and rejects all other encryption algorithms: [***aes128-ctr, aes256-ctr, aes256-gcm@openssh.com***].

NDcPP21:FCS_SSHS_EXT.1.5
The TSF shall ensure that the SSH public-key based authentication implementation uses [***ssh-rsa, rsa-sha2-256, rsa-sha2-512***] as its public key algorithm(s) and rejects all other public key algorithms. (TD0424 applied)

NDcPP21:FCS_SSHS_EXT.1.6
The TSF shall ensure that the SSH transport implementation uses [***hmac-sha1, hmac-sha1-96, hmac-sha2-256, hmac-sha2-512, AEAD_AES_256_GCM***] as its data integrity MAC algorithm(s) and rejects all other MAC algorithm(s).

NDcPP21:FCS_SSHS_EXT.1.7
The TSF shall ensure that [***diffie-hellman-group14-sha1, ecdh-sha2-nistp256***] and [***ecdh-sha2-nistp384, ecdh-sha2-nistp521***] are the only allowed key exchange methods used for the SSH protocol.

NDcPP21:FCS_SSHS_EXT.1.8
The TSF shall ensure that within SSH connections, the same session keys are used for a threshold of no longer than one hour, and each encryption key is used to protect no more than one gigabyte of data. After any of the thresholds are reached, a rekey needs to be performed. (TD0475 applied)

5.1.2.12 TLS Client Protocol with authentication (NDcPP21:FCS_TLSC_EXT.2)

NDcPP21:FCS_TLSC_EXT.2.1
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***].

NDcPP21:FCS_TLSC_EXT.2.2
The TSF shall verify that the presented identifiers of the following types [***identifiers defined in RFC 6125***] are matched to reference identifiers. (TD0481 applied)

NDcPP21:FCS_TLSC_EXT.2.3
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***].

NDcPP21:FCS_TLSC_EXT.2.4
The TSF shall [***present the Supported Elliptic Curves Extension with the following NIST curves: [secp384r1] and no other curves***] in the Client Hello.

NDcPP21:FCS_TLSC_EXT.2.5
The TSF shall support mutual authentication using X.509v3 certificates.

5.1.2.13 TLS Server Protocol with mutual authentication (NDcPP21:FCS_TLSS_EXT.2)

NDcPP21:FCS_TLSS_EXT.2.1
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***].

NDcPP21:FCS_TLSS_EXT.2.2
The TSF shall deny connections from clients requesting SSL 2.0, SSL 3.0, TLS 1.0, and [***TLS 1.1***].

NDcPP21:FCS_TLSS_EXT.2.3
The TSF shall [***generate EC Diffie-Hellman parameters over NIST curves [secp384r1] and no other curves***].

NDcPP21:FCS_TLSS_EXT.2.4
The TSF shall support mutual authentication of TLS clients using X.509v3 certificates.

NDcPP21:FCS_TLSS_EXT.2.5
When establishing a trusted channel, by default the TSF shall not establish a trusted channel if the client certificate is invalid. The TSF shall also [***Not implement any administrator override mechanism***].

NDcPP21:FCS_TLSS_EXT.2.6
The TSF shall not establish a trusted channel if the distinguished name (DN) or Subject Alternative Name (SAN) contained in a certificate does not match the expected identifier for the client.

5.1.3 Identification and authentication (FIA)

5.1.3.1 Authentication Failure Management (NDcPP21:FIA_AFL.1)

NDcPP21:FIA_AFL.1.1

The TSF shall detect when an Administrator configurable positive integer within [***0-9***] unsuccessful authentication attempts occur related to Administrators attempting to authenticate remotely using a password. (TD0408 applied)

NDcPP21:FIA_AFL.1.2

When the defined number of unsuccessful authentication attempts has been met, the TSF shall [***prevent the offending Administrator from successfully establishing remote session using any authentication method that involves a password until [unlocking the account through the command line] is taken by an Administrator, prevent the offending Administrator from successfully establishing remote session using any authentication method that involves a password until an Administrator defined time period has elapsed***]. (TD0408 applied)

5.1.3.2 Password Management (NDcPP21:FIA_PMG_EXT.1)

NDcPP21: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 [***15***] characters.

5.1.3.3 Protected Authentication Feedback (NDcPP21:FIA_UAU.7)

NDcPP21: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.

5.1.3.4 Password-based Authentication Mechanism (NDcPP21:FIA_UAU_EXT.2)

NDcPP21:FIA_UAU_EXT.2.1

The TSF shall provide a local [***password-based, SSH public key-based***] authentication mechanism to perform local administrative user authentication. (TD0408 applied)

5.1.3.5 User Identification and Authentication (NDcPP21:FIA_UIA_EXT.1)

NDcPP21:FIA_UIA_EXT.1.1

The TSF shall allow the following actions prior to requiring the non-TOE entity to initiate the identification and authentication process:

- Display the warning banner in accordance with FTA_TAB.1;
- [***no other actions***].

NDcPP21: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.

5.1.3.6 X.509 Certificate Validation (NDcPP21:FIA_X509_EXT.1/Rev)

NDcPP21: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***]
- The TSF shall validate the extendedKeyUsage field according to the following rules:

o 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.

o 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.

o 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.

o 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.

NDcPP21: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.

5.1.3.7 X.509 Certificate Authentication (NDcPP21:FIA_X509_EXT.2)

NDcPP21: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***].

NDcPP21: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***].

5.1.3.8 X.509 Certificate Requests (NDcPP21:FIA_X509_EXT.3)

NDcPP21: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, Organization, Organizational Unit, Country***].

NDcPP21: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 (NDcPP21:FMT_MOF.1/Functions)

NDcPP21:FMT_MOF.1.1/Functions

The TSF shall restrict the ability to [***modify the behaviour of***] the functions [***transmission of audit data to an external IT entity***] to Security Administrators.

5.1.4.2 Management of security functions behaviour (NDcPP21:FMT_MOF.1/ManualUpdate)

NDcPP21:FMT_MOF.1.1/ManualUpdate

The TSF shall restrict the ability to enable the functions to perform manual update to Security Administrators.

5.1.4.3 Management of security functions behaviour (NDcPP21:FMT_MOF.1/Services)

NDcPP21:FMT_MOF.1.1/Services

The TSF shall restrict the ability to enable and disable start and stop services to Security Administrators.

5.1.4.4 Management of TSF Data (NDcPP21:FMT_MTD.1/CoreData)

NDcPP21:FMT_MTD.1.1/CoreData

The TSF shall restrict the ability to manage the TSF data to Security Administrators.

5.1.4.5 Management of TSF data (NDcPP21:FMT_MTD.1/CryptoKeys)

NDcPP21/VPNGW10:FMT_MTD.1.1/CryptoKeys

The TSF shall restrict the ability to manage the cryptographic keys to Security Administrators.

5.1.4.6 Specification of Management Functions (NDcPP21:FMT_SMF.1)

NDcPP21: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;
- [***o Ability to start and stop services,***
o Ability to configure audit behavior,
oAbility to manage the cryptographic keys,
o Ability to configure the cryptographic functionality,
o Ability to configure the lifetime for IPsec SAs,
o Ability to re-enable an Administrator account,
o Ability to set the time which is used for time-stamps;
o Ability to configure NTP,
o Ability to configure the reference identifier for the peer;
o Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors,
o Ability to import X509v3 certificates to the TOE's trust store].

5.1.4.7 Restrictions on Security Roles (NDcPP21:FMT_SMR.2)

NDcPP21:FMT_SMR.2.1

The TSF shall maintain the roles: - Security Administrator.

NDcPP21:FMT_SMR.2.2

The TSF shall be able to associate users with roles.

NDcPP21: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 Protection of the TSF (FPT)

5.1.5.1 Protection of Administrator Passwords (NDcPP21:FPT_APW_EXT.1)

NDcPP21:FPT_APW_EXT.1.1

The TSF shall store administrative passwords in non-plaintext form. (TD0483 applied)

NDcPP21:FPT_APW_EXT.1.2

The TSF shall prevent the reading of plaintext administrative passwords. (TD0483 applied)

5.1.5.2 Protection of TSF Data (for reading of all pre-shared, symmetric and private keys) (NDcPP21:FPT_SKP_EXT.1)

NDcPP21:FPT_SKP_EXT.1.1
The TSF shall prevent reading of all pre-shared keys, symmetric keys, and private keys.

5.1.5.3 Reliable Time Stamps (NDcPP21:FPT_STM_EXT.1)

NDcPP21:FPT_STM_EXT.1.1
The TSF shall be able to provide reliable time stamps for its own use.

NDcPP21:FPT_STM_EXT.1.2
The TSF shall [***allow the Security Administrator to set the time, synchronise time with an NTP server***].

5.1.5.4 TSF testing (NDcPP21:FPT_TST_EXT.1)

NDcPP21: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: [***cryptographic known answer tests, pair-wise consistency tests, KDF tests and software integrity test***].

5.1.5.5 Trusted update (NDcPP21:FPT_TUD_EXT.1)

NDcPP21: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***].

NDcPP21: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***].

NDcPP21:FPT_TUD_EXT.1.3
The TSF shall provide means to authenticate firmware/software updates to the TOE using a [***digital signature mechanism***] prior to installing those updates.

5.1.6 TOE access (FTA)

5.1.6.1 TSF-initiated Termination (NDcPP21:FTA_SSL.3)

NDcPP21:FTA_SSL.3.1
The TSF shall terminate a remote interactive session after a Security Administrator-configurable time interval of session inactivity.

5.1.6.2 User-initiated Termination (NDcPP21:FTA_SSL.4)

NDcPP21:FTA_SSL.4.1
The TSF shall allow Administrator-initiated termination of the Administrator's own interactive session.

5.1.6.3 TSF-initiated Session Locking (NDcPP21:FTA_SSL_EXT.1)

NDcPP21:FTA_SSL_EXT.1.1
The TSF shall, for local interactive sessions, [***- lock the session - disable any activity of the user's data access/display devices other than unlocking the session, and requiring that the administrator re-authenticate to the TSF prior to unlocking the session***] after a Security Administrator-specified time period of inactivity.

5.1.6.4 Default TOE Access Banners (NDcPP21:FTA_TAB.1)

NDcPP21: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.

5.1.7 Trusted path/channels (FTP)

5.1.7.1 Inter-TSF trusted channel (NDcPP21:FTP_ITC.1)

NDcPP21: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, [***authentication server, [NTP server], external optical network peer***] 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.

NDcPP21:FTP_ITC.1.2

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

NDcPP21:FTP_ITC.1.3

The TSF shall initiate communication via the trusted channel for [

- ***audit server using IPSec;***
- ***authentication server using IPSec;***
- ***NTP server using IPSec;***
- ***external optical network peer using TLS;***].

5.1.7.2 Trusted Path (NDcPP21:FTP_TRP.1/Admin)

NDcPP21:FTP_TRP.1.1/Admin

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

NDcPP21:FTP_TRP.1.2/Admin

The TSF shall permit remote Administrators to initiate communication via the trusted path.

NDcPP21:FTP_TRP.1.3/Admin

The TSF shall require the use of the trusted path for initial Administrator authentication and all remote administration actions.

5.2 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 3 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 TSF shall support mutual authentication of TLS clients using X.509v3 certificates.

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
- Protection of the TSF
- TOE access
- Trusted path/channels

6.1 Security audit

6.1.1 FAU_GEN.1, FAU_GEN.2

The TOE generates audit records for start-up and shutdown of the TOE, all administrator actions, and for all the events identified in Table 2 Auditable Events. Audit records include date and time of the event, type of event, user identity that caused the event to be generated, the outcome of the event, as well as the additional content listed in column 3 of Table 2. The TOE also attaches the hostname to each audit record for easy distinction between different devices. The TOE audits any changes to the database configurations as well as the status of the command. The TOE also reports the username associated with commands that log the user's action (such as login, logout, and image upgrade). For cryptographic keys in the form of X.509 certificate private and public keys, the TOE assigns a unique name identifier to each certificate imported. The certificate is associated with an asymmetric key pair, which also has a unique name identifier (for example, if the certificate is identified as NECERT-3-1, then the corresponding keypair is ASYMKEY-3).

6.1.2 FAU_STG_EXT.1

The TOE is a standalone TOE that is able to transmit audit logs to an external syslog server over a secure IPSec channel. The TOE transfers audit logs to a syslog server in real-time. The TOE simultaneously stores audit logs locally and transmits the same logs remotely (within one second of each other). When the local audit storage is full, the TOE will delete the oldest log so that new events will appear at the top of the log. The TOE transmits audit records in real time over a secure IPSec channel, which encrypts the data before arriving at the IPSec peer. The TOE stores a maximum of 1000 logs in the local audit buffer. An authorized user must log into the TOE in order to view the local audit records.

6.2 Cryptographic support

The TOE includes two cryptographic modules that provide supporting cryptographic functions. The mTera TOE contains an OpenSSL FIPS object module and a kernel crypto module for cryptographic services. The Kernel Crypto module is used for IPSEC datapath (encrypt/decrypt, message authentication, hash) while the OpenSSL module is used for IPSEC Key management (i.e. IKE) and TLS operations. The OpenSSL module is used for secret negotiation, authentication, encrypt/decrypt, message authentication, hashes and DRBG services.

The evaluated configuration requires that the TOE be configured in FIPS mode to ensure that the CAVP tested algorithms are used. The following functions have been CAVP certified:

Requirements	Functions	Standards	Cert	Cert
	Cryptographic key generation		NXP QorIQ P4080 (Infinera OpenSSL Library)	NXP QorIQ P4080 (Linux 3.4 Kernel module)
FCS_CKM.1	RSA schemes using cryptographic key sizes of 2048-bit or greater	FIPS Pub 186-4	C537	**N/A**
FCS_CKM.1	ECC schemes using 'NIST curves' P-256, P-384 and P-521	FIPS Pub 186-4	C537	**N/A**
	Cryptographic key establishment/distribution			
FCS_CKM.2	Elliptic curve-based key establishment schemes	NIST SP 800-56A	C537	**N/A**
	Encryption/Decryption			
FCS_COP.1/ DataEncryption	AES CBC/CTR (128 and 256 bits)	ISO 18033-3 (AES) ISO 10116 (CBC and CTR mode)	C537	C538
FCS_COP.1/ DataEncryption	AES GCM (256 bits)	ISO 18033-3 (AES) ISO 19772 (GCM mode)	C537	C538
	Cryptographic signature services			
FCS_COP.1/SigGen	RSA Digital Signature Algorithm (rDSA) (modulus 2048)	FIPS Pub 186-4	C537	**N/A**
FCS_COP.1/SigGen	Elliptic Curve Digital Signature Algorithm (ECDSA) with an elliptical curve size of 384 bits	FIPS Pub 186-4	C537	**N/A**
	Cryptographic hashing			
FCS_COP.1/Hash	SHA-1/256/384/512 (digest sizes 160, 256, 384, 512 bits)	ISO/IEC 10118-3:2004	C537, A1646	C538, A1646
	Keyed-hash message authentication			
FCS_COP.1/ KeyedHash	HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512 (digest sizes 160, 256, 384, 512 bits, respectively)	ISO/IEC 9797-2:2011, Section 7 ‘MAC Algorithm 2’	C537, A1646	C538, A1646
	Random bit generation			

Requirements	Functions	Standards	Cert	Cert
FCS_RBG_EXT.1	CTR_DRBG (AES) with HW based noise sources (256 bits)	ISO/ICE 18031:2011	C537	**N/A**

Table 4 Cryptographic Functions

6.2.1 FCS_CKM.1

The TOE supports key generation using FIPS Pub 186-4 RSA (2048-bit) and ECDSA (P-384) when generating key pairs for certificate signing requests. For digital signature verification of peer certificates, the TOE uses either RSA or ECDSA for IPSec, ECDSA for TLS, and RSA for SSH. The TOE generates keys according to RFC 3526 for Diffie-Hellman group 14 for key exchange in IPSec sessions. The TOE also generates Elliptic Curve Diffie-Hellman keys for key exchange in TLS. The TOE generates 2048-bit RSA keys for key exchange in SSH.

6.2.2 FCS_CKM.2

The TOE uses ECDHE with P-384 as the key establishment algorithm in both TLS (FCS_TLSC_EXT.2 and FCS_TLSS_EXT.2) and IPSec (FCS_IPSEC_EXT.1). The TOE uses DH 2048 bit group 14 in IPsec (FCS_IPSEC_EXT.1) only. The TOE uses DH 2048 bit group 14 with SHA-1 as well as ECDHE with curve sizes P-256, P-384 and P-521 for SSH key exchange.

6.2.3 FCS_CKM.4

The TOE is designed to overwrite secret and private keys when they are no longer required by the TOE. Overwriting the keys is accomplished by overwriting the secret or private key with random data that is generated from the TOE's ISO/IEC 18031:2011 DRBG.

The TOE supports the following persistent keys:

Key Item	Key function	Key Generation Method	Key Output	Key Storage	Key Zeroization
X.509 auth.	X.509 certificates	Externally provided	CSR upload	SD Card with AES encryption	Zeroization on manual delete
Master key	Master key; Key to encrypt the other Key/CSP	Generated internally by DRBG	No output	EEPROM	FIPS/NONFIPS mode switching; Manual Zeroization
PID	User password	Input by TL1 command	No output	SD Card with AES encryption	FIPS/NONFIPS mode switching; Manual Zeroization
System private Key for TLS and IPsec – RSA	System private Key – RSA	RSA Private key for generation of signatures, authentication and key establishment; Generated through command; Used to export CSR;	No output	SD Card with AES encryption	FIPS/NONFIPS mode switching; Manual Zeroization;

		Associated with NE certificate			
System public Key for TLS and IPsec - RSA	System public Key - RSA	Generated from System private Key in running time if requested by software functions	output	SD Card with AES encryption	FIPS/NONFIPS mode switching; Manual Zeroization; Power cycle
System private Key for TLS and IPsec - ECDSA	System private Key - ECDSA	ECDSA Private key for generation of signatures, authentication and key establishment; Generated through command; Used to export CSR; Associated with network element (NE) certificate	No output	SD Card with AES encryption	FIPS/NONFIPS mode switching; Manual Zeroization;
System public Key for TLS and IPsec ECDSA	System public Key - ECDSA	Generated from System private Key in running time if requested by software functions	output	SD Card with AES encryption	FIPS/NONFIPS mode switching; Manual Zeroization; Power cycle
SSHv2 server private key - ECDSA	SSH Key	Generated through command (ED-TCPIP)	No output	SD Card with AES encryption	FIPS/NONFIPS mode switching; Manual Zeroization;
SSHv2 server public key - ECDSA	SSH public Key	Generated through command (ED-TCPIP)	output	SD Card with AES encryption	FIPS/NONFIPS mode switching; Manual Zeroization;
SSHv2 server private key - RSA	SSH Key	Generated through command (ED-TCPIP)	No output	SD Card with AES encryption	FIPS/NONFIPS mode switching; Manual Zeroization;
SSHv2 server public key - RSA	SSH public Key	Generated through command (ED-TCPIP)	output	SD Card with AES encryption	FIPS/NONFIPS mode switching; Manual Zeroization;
RADIUS shared secret	RADIUS shared secret	Input by TL1 command	No output	SD Card with AES encryption	FIPS/NONFIPS mode switching; Manual Zeroization;
Network element (NE) local certificate for TLS and IPsec(including	NE local certificate (including System public Key) Key - ECDSA	Downloaded from external file server	No output	SD Card with AES encryption	FIPS/NONFIPS mode switching; Manual Zeroization; Power cycle

System public Key) Key - ECDSA					
NE local certificate for TLS and IPsec(including System public Key) -RSA	NE local certificate (including System public Key) -RSA	Downloaded from external file server	output	SD Card with AES encryption	FIPS/NONFIPS mode switching; Manual Zeroization; Power cycle
CA certificate for TLS and IPsec	CA certificate - RSA	Downloaded from external file server	No output	SD Card with AES encryption	FIPS/NONFIPS mode switching; Manual Zeroization; Power cycle
CA certificate for TLS and IPsec	CA certificate - ECDSA	Downloaded from external file server	No output	SD Card with AES encryption	FIPS/NONFIPS mode switching; Manual Zeroization; Power cycle
Data integrity check - public RSA key	Data integrity check - public RSA key	hardcoded in the software image	No output	SD Card with AES encryption	

6.2.4 FCS_COP.1/DataEncryption

The TOE supports AES CBC (128 and 256 bits) and AES GCM (256 bits) for data encryption/decryption in IPSec. The TOE supports AES CTR (128 and 256 bits) and AES GCM (128 and 256 bits) for data encryption/decryption in SSH. The TOE supports AES 256 GCM for data encryption/decryption in TLS.

6.2.5 FCS_COP.1/Hash

The TOE supports SHA-1/256/384/512 (digest sizes 160, 256, 384, and 512 bits) for cryptographic hashing. Hash functions are used in RSA/ECDSA digital signature generation/verification as well as in the key exchange algorithms for IPSec, TLS and SSH.

6.2.6 FCS_COP.1/KeyedHash

The TOE supports the following keyed hash algorithms:

	Key Length (bits)	Block size (bits)	Output MAC length (bits)
HMAC-SHA-1-96	160	160	96
HMAC-SHA2-256	256	256	128
HMAC-SHA2-384	384	384	192
HMAC-SHA2-512	512	512	256

6.2.7 FCS_COP.1/SigGen

The TOE supports RSA (modulus 2048) and ECDSA with elliptical curve size 384 bits for signature generation and verification.

6.2.8 FCS_IPSEC_EXT.1

The TOE can be configured with SPD rules that will either bypass, protect, or discard traffic based on IP addresses and port numbers. The TOE's ENT-SPD command allows configuration of IP protocol number, local IP address and port number, remote IP address and port number, and the rule action, which are protect, bypass or discard. If the protect action is specified, the user can configure the protection mode (tunnel/transport), the ciphersuite used for protection, and rekeying lifetimes in either bytes or based on time. The SPD priority is hardcoded into the TOE's firmware. The TOE processes Bypass rules first, Protect rules second, and then Discard rules last. The TOE can be configured to have a default discard rule that will discard all traffic that does not match any other SPD rule. If there are multiple rules within each action, the TOE processes the lowest order first. For example, if SPD-1-1-100 and SPD-1-1-101 both specify a BYPASS rule, the TOE processes SPD-1-1-100 first. If the traffic does not match, then the TOE moves on to SPD-1-1-101.

The TOE supports IPSec according to RFC 4301. The TOE supports IKEv2 IPSec in both tunnel and transport mode.

The TOE supports AES-CBC 128/256 bits (specified by RFC 3602) and AES-GCM 256-bit encryption algorithms for both IKEv2 and ESP, with HMAC-SHA-1 and HMAC-SHA-384 for integrity in both IKEv2 and ESP.

IKEv2 and IKEv2 Child SA lifetimes in the TOE can be configured with a time value up to 24 hours. The TOE's IKEv2 Child SA lifetime can also be configured with a limit based on the number of bytes.

The TOE also supports both DH-2048 (group 14) and ECDHE-384 (group 20), both of which can be configured. In the IKEv2 IKE_SA and IKE_CHILD exchanges, the TOE and peer will agree on the best DH group both can support. When the TOE initiates the IKE negotiation, the DH group is sent in order according to the peer's configuration. When the TOE receives an IKE proposal, it will select the first match and the negotiation will fail if there is no match.

The TOE accepts either a 128-bit or 256-bit key strength algorithm for both IKE and ESP. The TOE also checks for the strength of the IKE versus the ESP algorithms. The TOE first blocks any attempts to set an ESP algorithm that is stronger than the IKE algorithm, and then the TOE does a secondary check to ensure that the IKE algorithm is at least as strong as the ESP algorithm.

The TOE supports both RSA and ECDSA for authentication in IPSec. The TOE does not support pre-shared keys for authentication.

The TOE allows configuration of an expected peer identifier in the form of a Distinguished Name. If the expected DN and the DN in the peer's certificate do not match, the TOE will disconnect and report a failure. Fields within the DN are not individually selectable; the DN must be an exact match for the entire DN string.

The value of x in both Diffie-Hellman and EC-Diffie-Hellman is 384 bits. The TOE supports the following PRF hash functions: PRF_HMAC_SHA1 and PRF_HMAC_SHA384 and generates nonces used in the IKEv2 exchanges of at least 128 bits in size and at least half the output size of the negotiated pseudorandom function (PRF) hash. The nonces generated are 32 bytes (256 bits) in length. All values are generated using the TOE's ISO/IEC 18031:2011 AES-256 CTR DRBG.

6.2.9 FCS_NTP_EXT.1

The TOE can synchronize its time with an external NTP server using the NTPv4 protocol. The TOE must establish a successful IPSec connection between itself and the NTP server. The IPSec session ensures that data is encrypted and that the two peers are authenticated before data is transmitted. The TOE does not update the NTP from broadcast or multicast addresses. The TOE's configuration allows adding at least 3 NTP time sources.

6.2.10 FCS_RBG_EXT.1

The TOE contains an NXP QorIQ P4080 for cryptographic operations. The TOE supports an AES-CTR 256-bit ISO/IEC 18031:2011 DRBG. The DRBG is seeded with entropy from an onboard noise source. The hardware noise source is a Kinetis K82 true RNG (TRNG) chip that meets ISO/IEC 18031:2011 Table C1 'Security Strength Table for Hash Functions' specifications. Entropy from the K82 chip is made available via an I2C driver, which is read by the Entropy Gathering Daemon (EGD) and made available to user-mode applications. The calculated min entropy is 384-bits when the entropy-delay is at the right value. The mTera firmware contains a check that ensures that the hardware based noise source is providing full entropy.

6.2.11 FCS_SSHS_EXT.1

The TOE complies with the following SSHv2 RFCs: 4251, 4252, 4253, 4254, 5656, 6668, 8332. The TOE supports both password-based and public key-based authentication methods in SSHv2. The TOE checks the packet length of an incoming packet. If a packet that is greater than 262144 bytes sent to the TOE, the TOE drops the packet and terminates the SSH session. The TOE supports the following algorithms:

Encryption: aes128-ctr, aes256-ctr, aes128-gcm@openssh.com, aes256-gcm@openssh.com

Integrity: hmac-sha1, hmac-sha1-96, hmac-sha2-256, hmac-sha2-512, AEAD_AES_128_GCM, AEAD_AES_256_GCM

Authentication: ssh-rsa, rsa-sha2-256, rsa-sha2-512

Key Exchange: diffie-hellman-group14-sha1, ecdh-sha2-nistp256, ecdh-sha2-nistp384, ecdh-sha2-nistp521

The TOE automatically initiates a rekey of the SSH session keys at either 1 hour or 1 gigabyte of traffic, whichever comes first.

6.2.12 FCS_TLSC_EXT.2, FCS_TLSS_EXT.2

The TOE supports TLSv1.2 sessions between itself and an external optical network peer. The data transmitted is network communications traffic that is relayed from one TOE to another TOE. The TOE only supports the TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 ciphersuite. Per FIA_X509_EXT.2.1, the TOE supports mutual authentication of a peer certificate. The TOE also performs X.509 security checks on a peer certificate. If the certificate is deemed invalid, the TOE will terminate the connection. The TOE supports checking of an expected reference identifier against the peer certificate's reference identifier. An administrator can set the expected reference identifier when setting up the system. The TOE checks for a SAN:URI field and an FQDN either in the Common Name or SAN:DNS. IP addresses are not supported. The TOE implements wildcard checking in both the CN and SAN only. The TOE checks the SAN before checking the CN. If the identifier check fails, the TOE will terminate the session establishment attempt. The TOE only supports the P-384 (secp384r1) curve during the ECDHE key exchange, and this value is not configurable.

The TOE also requires that a certificate is configured for the TLS object. This is done through the ENT-ENCAPP command, which takes a certificate and ties it to the TLS object.

6.3 Identification and authentication

Prior to requiring the non-TOE entity to initiate the identification and authentication process, the TOE displays an Authorized Administrator-specified advisory notice and consent warning message regarding unauthorized use of the TOE (FTA_TAB.1). The TOE requires an administrator to be successfully identified and authenticated before allowing any other TSF-mediated actions on behalf of that user.

6.3.1 FIA_AFL.1

The TOE tracks each unsuccessful authentication attempt for each account. Once the account is authenticated successfully and before the counter reaches the limit for failed authentications, the counter is reset. If the counter reaches the configured limit (0-9), then the account will be locked out and prevented from gaining access even if a

valid password is provided. The TOE allows configuration of the lockout behavior to be either through an administrator unlock command or until a timer has expired. The account is thereby unlocked upon expiration of the lockout period, or if an administrator has issued an unlock command.

6.3.2 FIA_PMG_EXT.1

The TOE allows administrators to configure passwords for users that are between 8 and 15 characters. Along with upper and lower case letters and numbers, the TOE allows the following special characters: '!', '@', '#', '$', '%', '^', '&', '*'. The TOE provides obscured feedback to the administrative user while the authentication is in progress at the local console. The TOE can be configured with either a local database or remote RADIUS database for authenticating users.

6.3.3 FIA_UAU.7, FIA_UAU_EXT.2, FIA_UIA_EXT.1

The TOE allows users to log into the TOE using SSHv2 locally or remotely. In local SSHv2 sessions, the LCI interface is not routed, therefore a user must be directly connected to the LCI interface of the TOE. In remote SSHv2 sessions, the DCN interface is routed, so users can access the TOE through remote means. Both local and remote SSH support password and public key authentication. If a user successfully logs in, the TOE presents the TL1 command-line prompt. An example of the prompt is below:

MTERA_1>

In the example, the “MTERA_1” is the hostname of the device, and this will be different for each system. Before a user is logged on via SSHv2, the only action available is to view the login banner according to FTA_TAB.1.

The TOE has a dual authentication mechanism. After logging in through SSHv2 to get to the command line prompt, the operator must also authenticate one more time in order to access all commands. The TOE allows the user to list commands available by typing a question mark at the command line, however the user cannot execute any of the commands. The only command available to the operator at this point is the ACT-USER command. After the operator executes the ACT-USER command and provides a password, the operator has access to all commands available at the user’s privilege level.

The TOE also supports authentication using an external server. The TOE communicates with a RADIUS server for authentication of configured users. The communication with the RADIUS server is protected via IPSec. The user can configure a RADIUS server using the following command:

ent-aaa:::ctag:::role=auth,server=<IP ADDRESS>, PORT=<PORT NUMBER>, SECRET=<SECRET WORD OR PHRASE>, proto=RADIUS

6.3.4 FIA_X509_EXT.1, FIA_X509_EXT.2, FIA_X509_EXT.3

During configuration of the trusted channel (either TLS or IPSec), an administrator must specify the certificate to be used with the trusted channel. This is done with the ENT-ENCAPP or ED-ENCAPP (ENT is for creating a new object, ED is for editing an existing object) command, which defines one of the TOE’s encryption application and specifies a certificate that the TOE sends as part of mutual authentication with the peer in both TLS and IPSec. The ENCAPP has a corresponding ENCPEER, which contains the peer’s connection details.

The TOE also checks the validity of the certificates it receives from its peers (such as the certificate's extended key usage, expiration, revocation, basic constraints, and reference identifiers) as part of the authentication step of TLS and IPSec connections. The TOE performs these checks on the peer's certificate before moving up the chain to check each intermediate CA in the chain for revocation and basic constraints. If the certificate is invalid, the TOE rejects the connection attempt. For both TLS and IPSec, the TOE relies on certificate revocation checking by communicating with an external server. If the TOE cannot establish communications with the external server for revocation checks, then the TOE will terminate the connection attempt with the peer. This applies to both IPSec and TLS channels.

The TOE checks the TLS/IPSec peer's certificate revocation during the authentication step of a TLS/IPSec connection. The TOE checks each level of the certificate chain sent by the peer. If any certificate in the chain is revoked, the TOE will reject the connection attempt. The TOE supports CRLs, and it will download CRLs using the the CDP URL in the peer's certificate.

The TOE generates Certificate Request Messages and includes the following information: public key, common name, organization, organizational unit, country. Upon receiving the CA Certificate response, the TOE will validate the chain of certificates from the Root CA.

6.4 Security management

The TOE maintains the security role of Security Administrator who can manage the TOE both remotely and locally. The TOE supports role-based authentication. Administrators can make use of both local and remotely accessible administrator interfaces.

All roles authenticate with a username and password via the interactive command line. Local administrators can also use the CLI via a direct connection to the TOE's LCI interface by using username and password. Although the local administration is through SSH, the LCI interface is not routed, meaning users must directly connect to the LCI interface. Remote administrators may use the CLI interface via an SSH protocol connection from an SSH client through the TOE's DCN interface.

6.4.1 FMT_MTD.1/CoreData

Only Security Administrators can manage TSF data. There are no security functions available through any interfaces prior to administrator login. Non-administrative users do not have access to the TOE via the CLI, therefore, they do not have any access to the security functions of the TOE. The TOE employs a dual authentication mechanism. The user must authenticate via SSH to access the TL1 command line. At this point, the user only has access to the ACT-USER command. The user must execute the command, which requires a password to authenticate the user, in order to access the command set for the active user.

Each user is assigned a privilege level during user creation. The user level determines the services available at each stage. A user has access to the commands at and below their designated level.

The following table shows the commands available for the different system access levels.

Table 6.2 Security and Administration User Privileges

Public (A2)	Test (A4)	Provisioning (A6)	Operator (A7)	Admin (A8) & EMS
ACT-USER	OPR-ACO-ALL	ALW-PMFILE-{X}	ALW-BKUPSCHED-MEM	ALW-PKT-SNMPV2
ALW-MSG-ALL	OPR-ARC-{X}	ALW-PMREPT-{X}	DLT-DB	COPY-RFILE
CANC-USER	OPR-CABL-DETECT	DLT-{X} (facilities)	DLT-FTPSERVER	DLT-AAA
ED-PID	OPR-EXT-CONT	DLT-BL	DLT-IPPG	DLT-SESSION
INH-MSG-ALL	OPR-FINDRTE	DLT-CONN	DLT-NTPPEER	DLT-USER-SECU
MEAS-OPTPWR	OPR-LPBK-{X}	DLT-CRS-OCH	DLT-RFILE	ED-AAA
RTRV-{X} (facilities)	OPR-PROTNSW-{X}	DLT-CRS-ODUk	DLT-TRAPIP	ED-NE
RTRV-ALM-{X}	RLS-ARC-{X}	DLT-DA	ED-ALMPF-{X}*	ED-PROXY
RTRV-ALM-ENV	RLS-EXT-CONT	DLT-EQPT	ED-BL	ED-SECU-SYS
RTRV-ALMPF-{X}*	RLS-LPBK-{X}	DLT-EXDPATH	ED-BL-MEMBER	ED-TCPIP
RTRV-ARC-{X}*	RLS-PROTNSW-{X}	DLT-EXPPATH	ED-CALL	ED-USER-SECU
RTRV-ATTN-OCH	RTRV-USER-SECU	DLT-FFP-HGE	ED-CPPF	ED-WARNING
RTRV-ATTR-CONT	STA-BER	DLT-FFP-OCn	ED-DB	ENT-AAA
RTRV-ATTR-ENV	STP-BER	DLT-FFP-STMn	ED-FTPSERVER	ENT-USER-SECU
RTRV-BER		DLT-FFP-TGLAN	ED-IP	INH-PKT-SNMPV2
RTRV-BKUPSCHED-MEM		DLT-FIBR-EQPT	ED-IPPG	RTRV-AAA
RTRV-BL		DLT-GCC	ED-LINKPF	RTRV-ALMGEN
RTRV-BL-MEMBER		DLT-MGTETH	ED-NTPPEER	RTRV-LOG
RTRV-CALL		DLT-OCH	ED-PMPF	RTRV-SESSION
RTRV-CALL-DETAIL		DLT-TSL	ED-PPPPF	RTRV-USER-SECU
RTRV-CEF		ED-{X} (facilities)	ED-RSVPADJ	STA-ALMGEN
RTRV-COND-{X}		ED-CRS-OCH	ED-SLPF	STP-ALMGEN
RTRV-CONN		ED-CRS-ODUk	ED-SLPOLICY	ENT-DN
RTRV-CONN-INFO		ED-DA	ED-SNMP	DLT-DN
RTRV-CONN-ROUTE		ED-EQPT-{X}	ED-SNMP-COMMPREFIX	RTRV-DN
RTRV-CPPF		ED-EXDPATH	ED-STAT-RTE	ENT-ASYMKEY
RTRV-CRS-OCH		ED-EXPPATH	ENT/DLT-RSVP	DLT-ASYMKEY
RTRV-CRS-ODUk		ED-FFP-HGE	ENT/DLT-STAT-RTE	RTRV-ASYMKEY
RTRV-DGNCP		ED-FFP-OCn	ENT/ED/DLT-NP	OPR-EXPORT-CSR
RTRV-DA		ED-FFP-STMn	ENT/ED/DLT-OSCX	ENT-CERT
RTRV-DB		ED-FFP-TGLAN	ENT/ED/DLT-OSPF	ED-CERT
RTRV-DISP-OCH		ED-FGE	ENT/ED/DLT-OSPFADJ	DLT-CERT
RTRV-ENGIDMAP		ED-GCC	ENT/ED/DLT-OSPFAREA	RTRV-CERT
RTRV-EQPT		ED-MGTETH	ENT-FTPSERVER	ENT-ENCAPP
RTRV-ETH		ED-OCH	ENT-IPPG	ED-ENCAPP
RTRV-EXDPATH		ED-OMS	ENT-NTPPEER	DLT-ENCAPP
RTRV-EXPPATH		ED-OSC	ENT-STAT-RTE	RTRV-ENCAPP
RTRV-EXT-CONT		ED-OSPFRP	ENT-TRAPIP	ENT-ENCPEER
RTRV-FFP-HGE		ED-OTS	INH-BKUPSCHED-MEM	ED-ENCPEER
RTRV-FFP-OCn		ED-PPG-ODUk	INIT-SYS	DLT-ENCPEER
RTRV-FFP-STMn		ED-SLOT	INSTALL-CEF	RTRV-ENCPEER
RTRV-FFP-TGLAN		ED-TSL	INSTALL-SW	RTRV-ENCPEERADJ
RTRV-FIBR-EQPT		ED-WCG	OPR-PING	ENT-ENCODU
RTRV-FTPSERVER		ENT-BL	OPR-TRACE-ROUTE	ED-ENCODU
RTRV-GAIN-OTS		ENT/DLT-CALL	OPR-UPG-ABORT	DLT-ENCODU
RTRV-GCC		ENT-DA	OPR-UPG-COMMIT	RTRV-ENCODU
RTRV-HDR		ENT/DLT-OSPFRP	RTRV-SECU-SYS	ENT-SPD
RTRV-INTIP-SLOT		ENT/DLT-OSPFRPMAP		ED-SPD
RTRV-INV		ENT/DLT-TNALNKMAP		RTRV-SPD
RTRV-IP		ENT/ED/DLT-NODE		DLT-SPD
RTRV-IPPG		ENT/ED/DLT-TL		OPR-FIPS-ZEROIZECSP
RTRV-LADJ-TL		ENT-{X} (facilities)		
RTRV-LEDS		ENT-CRS-OCH		
RTRV-LINKPF		ENT-CRS-ODUk		
RTRV-MGTETH		ENT-EQPT		
RTRV-NE				

Table 6.2 Security and Administration User Privileges (Continued)

Public (A2)	Test (A4)	Provisioning (A6)	Operator (A7)	Admin (A8) & EMS
RTRV-NETYPE RTRV-NODE RTRV-NP RTRV-NP-STATS RTRV-NTPPEER RTRV-OCH RTRV-OMS RTRV-OPEDATA RTRV-OPTPWR-OTS RTRV-OSC RTRV-OSCX RTRV-OSPF RTRV-OSPFADJ RTRV-OSPFAREA RTRV-OSPFRP RTRV-OSPFRPMAP RTRV-OTS RTRV-PM-{X} RTRV-PMDAY RTRV-PMFILESCHED-{X} RTRV-PMMODE-{X} RTRV-PMPF RTRV-PMPFUSE RTRV-PMSCHED-{X}* RTRV-PPG-ODUk RTRV-PPPPF RTRV-PROXY RTRV-PTHTRC-OCH RTRV-PTHTRC-ODUF RTRV-PTHTRC-ODU0 RTRV-PTHTRC-ODU1 RTRV-PTHTRC-ODU2 RTRV-PTHTRC-ODU2E RTRV-PTHTRC-ODU3 RTRV-PTHTRC-ODU4 RTRV-PTHTRC-ODU RTRV-PTHTRC-OTUC2 RTRV-PTHTRC-OTUC3 RTRV-PTHTRC-OTS RTRV-RFILE RTRV-RSVP RTRV-RSVPADJ RTRV-RTE-ALL		ENT-EXDPATH ENT-EXPPATH ENT-FFP-HGE ENT-FFP-OCn ENT-FFP-STMn ENT-FFP-TGLAN ENT-FGE ENT-FIBR-EQPT ENT-GCC ENT-MGTETH ENT-OCH ENT-TSL INH-PMFILE-{X} INH-PMREPT INIT-REG-{X} OPR-ADMREROUTE OPR-CPSW RLS-CPSW RTRV-FPGAVERMAP RTRV-FPGAVER RTRV-USER-SECU SCHED-PMFILE-{X} SCHED-PMREPT-{X} SET-ATTN-OCH SET-GAIN-OTS SET-OPTPWR-OTS SET-OPTTH-OCH SET-PMDAY SET-PMMODE-{X} SET-TCAMODE-{X} SET-TH-{x} SW-DX	OPR-UPG-EXECUTE OPR-UPG-PATCHAPPLY RTRV-AO RTRV-SNMP-COMM RTRV-SNMP-COMMPREFIX RTRV-TCPIP RTRV-USER-SECU SCHED-BKUP-MEM SET-ATTR-CONT SET-ATTR-ENV SET-DAT SET-SID	

Table 6.2 Security and Administration User Privileges (Continued)

Public (A2)	Test (A4)	Provisioning (A6)	Operator (A7)	Admin (A8) & EMS
RTRV-SLOT RTRV-SLPF RTRV-SLPOLICY RTRV-SNMP RTRV-STAT-RTE RTRV-SW RTRV-SWVER RTRV-SWVERMAP RTRV-TCA-{X} RTRV-TCAMODE-{X} RTRV-TCE-TL RTRV-TH-{X} RTRV-TL RTRV-TNALNKMAP RTRV-TOD RTRV-TRAPIP RTRV-TSL RTRV-UPG-STATE RTRV-USER-SECU RTRV-WARNING RTRV-WCG				

A8 users have the highest privilege and are considered system admins. Administrators have access to TOE configuration editing commands. For example, administrators are the only ones with access to the ENT-CERT command, which is used to configure X.509 certificates used for mutual authentication, as well as configure trust anchor certificates in the TOE. Administrators can also use DLT-CERT to delete certificates, ED-CERT to edit existing certificate objects, and RTRV-CERT to retrieve information on current existing certificate objects.

6.4.2 FMT_SMF.1

The TOE provides the administrator with capabilities to manage all security functions identified in this Security Target, including the following:

- 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 configure audit behavior

- Ability to manage the cryptographic keys

- Ability to configure the cryptographic functionality

- Ability to configure the lifetime for IPsec SAs

- Ability to re-enable an Administrator account

- 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

Administrators are able to configure the TOE using the TL1 command line, which is accessed via SSHv2 using a local or remote connection. All commands for administering the TOE are available in both the local and remote SSHv2 connections.

6.5 Protection of the TSF

6.5.1 FPT_APW_EXT.1, FPT_SKP_EXT.1

The TOE encrypts passwords with AES-256 CBC before storing them in flash.

The TOE encrypts private and symmetric keys with AES-256 CBC before storing them in flash.

The TOE provides no interfaces to read pre-shared, symmetric keys, private keys or passwords.

6.5.2 FPT_STM_EXT.1

The TOE's system clock can be manually set or configured to sync with an external NTP source. Note that the clock is used primarily to provide a timestamp for audit records, but is also used to support timing elements of cryptographic functions, certificate validity checks, session timeouts, and unlocking of administrator accounts locked as a result of authentication failure.

6.5.3 FPT_TST_EXT.1

The TOE offers a suite of self-tests to verify the correct operation of the key generation and static TSF cryptographic data. Software images are cryptographically signed, and an image with an invalid signature will not be copied by the TOE into the image partition. If a self-test fails, the TOE will immediately halt operation and enter an error state thereby preventing potentially insecure operations (i.e., maintaining a secure state). The controller will reboot after a self-test failure. If a self-test failure continues to occur, the controller will continue to reboot repeatedly and will require return to manufacturer.

The following tests are performed:

Linux Kernel:

- AES-ECB (128/192/256) Encrypt/Decrypt KAT
- AES-CBC (128/192/256) Encrypt/Decrypt KAT
- AES-GCM (256) Encrypt/Decrypt KAT
- HMAC-SHA-1/SHA-256 KAT
- SHA-1/SHA-256 KAT

Strongswan:

- IKEv2 KDF test

OpenSSL:

- AES-ECB (128/256) Encrypt/Decrypt KAT
- AES-CBC (128/256) Encrypt/Decrypt KAT
- AES-GCM (256) Encrypt/Decrypt KAT
- DHE 2048 bits KAT

- DRBG KAT with health test
- ECDSA Pair-Wise Consistency test
- ECDH P-256/384/521 KAT
- ECDHE P-256/384/521 KAT
- HMAC-SHA1/ HMAC-SHA-256/384/512 KAT
- Key Wrapper AES-ECB (256) Encrypt/Decrypt KAT
- RSA Pair-Wise Consistency test
- SHA-1/ SHA-256/384/512 KAT

SNMP KDF test

TLS KDF test

SSH KDF test

AES-GCM (256) Encrypt/Decrypt KAT (Line card)

Software images integrity test with CRC32 (Main controller/Line card)

The TOE executes both known answer self-tests and pair-wise consistency tests during power-up. The known answer tests involve inputting known data into each algorithm and comparing the outputs against expected results. For example, the TOE's AES-CBC Encrypt KAT takes a known key, encrypts known plaintext using AES-CBC, and compares the result against known ciphertext. The KAT fails if the comparison results in the calculated ciphertext not matching the known ciphertext. For the Pair-Wise consistency test, the TOE takes a public/private key pair to calculate and verify a digital signature. If a pair-wise consistency test fails, the TOE shuts down the data output interface. If all self-tests pass, the TOE proceeds to normal operation.

6.5.4 FPT_TUD_EXT.1

An administrator can query the current version of the TOE by issuing the RTRV-SW command at the TL1 command-line prompt. The administrator first obtains the update file after contacting the vendor for a download link. The TOE's update process begins with an administrator uploading a valid image file. The administrator can issue the RTRV-SW command once more to view the path to the update file. The file's digital signature is verified before it is saved to the TOE's flash. The TOE checks the update image integrity using RSA-2048 with SHA-256. If the integrity verification fails, the TOE does not save the uploaded image to flash. Once the file is verified successfully and saved to the flash, the administrator can issue a command (either opr-upg-execute or opr-upg-patchapply) to install the image on the standby management card. The distinction between the two commands is that opr-upg-execute is for upgrading the TOE's entire image, and opr-upg-patchapply is for applying patches to the current existing software. This will apply the update to the standby card. Once that update is complete, the standby card will reboot itself automatically, and then it will become the active card. The previously active card will then apply the update, and it will also reboot itself to complete the update. Once the update is done applying, the administrator uses the opr-upg-commit to ensure the update is confirmed and saved. An important note is that upon the TOE reboot and boot stage, the image integrity is checked via a digital signature verification. If valid, the system will be allowed to restart on the latest image. If the image is not valid, the system will restart using the older image.

6.6 TOE access

6.6.1 FTA_SSL.3, FTA_SSL.4, FTA_SSL_EXT.1

The TOE terminates remote or local administrator sessions after session inactivity time exceeds a configurable session idle timeout. The session idle timeout is the maximum amount of time an administrator may remain idle.

All users may also terminate their own sessions at any time simply by logging off their session.

6.6.2 FTA_TAB.1

Whether connecting to the CLI remotely or locally, the TOE displays an advisory message when an administrator logs on. The administrator can configure the warning message displayed in the banner.

6.7 Trusted path/channels

6.7.1 FTP_ITC.1

The TOE protects communications between itself and external authentication (RADIUS), syslog, and NTP servers using IPSec/IKE. The protocol ensures that communication is protected. In this case, the TOE can either initiate IKE sessions or respond to IKE INIT requests from IPSec peers. The TOE also uses TLS to protect communications between itself and an external optical network peer. In this case, the TOE is either a client or server.

If an IPSec channel is unintentionally broken, the TOE treats this as an unrecoverable connection, so the TOE always sets up a new IKE SA.

For TLS, the TOE utilizes heartbeats (sent every second) to determine the status of an established session. If the TOE detects that a session has been disconnected, it will treat the session as unrecoverable and proceed to attempt a new TLS session establishment with a full handshake.

6.7.2 FTP_TRP.1/Admin

The TOE uses SSHv2 to secure communications between itself and authorized security 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 detection of modification of the communicated data. The protocol allows administrators to initiate communications via the trusted path.