Infinera GX G42 Optical Network Platform running Converged OS (COS) 6.2.10 Security Target

Version 1.1
01/15/2025

Prepared for:

Infinera Corporation

9005 Junction Dr, Suite C
Annapolis Junction, MD 20701

Prepared By:

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 the GX G42 Optical Network Platform running Converged OS (COS) version 6.2.10 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.

Acronyms

AES	Advanced Encryption Standard
ACVP	Automated Cryptographic Validation Protocol
AIA	Authority Information Access
CA	Certificate Authority
CAVP	Cryptographic Algorithm Validation Program
CC	Common Criteria
CCTL	CC Testing Laboratory
CDP	CRL Distribution Point
COS	Converged OS – Infinera's converged network operating system

CRL	Certificate Revocation List
CSPs	Critical Security Parameters
CSR	Certificate Signing Request
DH	Diffie-Hellman
DHE	Diffie-Hellman Ephemeral
DoD	Department of Defense
DSA	Digital Signature Algorithm
DWDM	Dense Wavelength Division Multiplexing
ECC	Elliptic Curve Cryptography
ECDHE	Elliptic Curve Diffie-Hellman Ephemeral
ECDSA	Elliptic Curve Digital Signature Algorithm
ESP	Encapsulating Security Payload
FFC	Finite Field Cryptography
FIPS	Federal Information Processing Standards
HTTPS	Hypertext Transfer Protocol Secure
ICE6, ICE7	Infinite Capacity Engine – Infinera's sixth and seventh generation Infinite Capacity Engines.
IKE	Internet Key Exchange
IPsec	Internet Protocol Security
NDcPP22e	collaborative Protection Profile for Network Devices, Version 2.2e, 23 March 2020
NEBS	Network Equipment-Building system
NETCONF	Network Configuration Protocol
NIAP	National Information Assurance Partnership
NIST	National Institute of Standards and Technology
NIT	Network iTC Interpretations Team
NSA	National Security Agency
NTP	Network Time Protocol
OCSP	Online Certificate Status Protocol
OE	Operational Environment
OS	Operating System
OTN	Optical Transport Network
PBKDF2	Password-Based Key Derivation Function 2
PKCS	Public Key Cryptography Standard
PP	Protection Profile
RADIUS	Remote Authentication Dial-In User Service

RBG	Random Bit Generator
RESTCONF	Representational State Transfer Configuration Protocol
RSA	Algorithm developed by Rivest, Shamir and Adleman
SA	Security Association
SAR	Security Assurance Requirement
SFR	Security Functional Requirement
SHA	Secure hash algorithm
SSHv2	Secure Shell 2.0
ST	Security Target
Syslog	System Logging Protocol
TACACS+	Terminal Access Controller Access Control Server
TCP	Transport Control Protocol
TD	Technical Decision
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Functions
TSS	TOE Summary Specification

1.1 Security Target Reference

ST Title – Infinera GX G42 Optical Network Platform running Converged OS (COS) 6.2.10 Security Target

ST Version – Version 1.1

ST Date – 01/15/2025

1.2 TOE Reference

TOE Identification – Infinera GX G42 Optical Network Platform running Converged OS (COS) 6.2.10

TOE Developer – Infinera Corporation

Evaluation Sponsor – Infinera Corporation

1.3 TOE Overview

The Target of Evaluation (TOE) is Infinera GX G42 Optical Network Platform running Converged OS (COS) version 6.2.10.

The GX G42 is an optical network platform delivering Wavelength, High-capacity Electrical OTN, and Packet network device. The GX G42 is equipped with four service slots in 3RU. This carrier-grade platform offers full Network Equipment-Building System (NEBS) Level 3 compliance, redundant controllers, multi-chassis management (not tested in this evaluation), and enhanced programmability. It offers high capacity and low power consumption by leveraging Infinera's 1.6T (2 x 800G per wavelength) sixth-generation Infinite Capacity Engine (ICE6) and Infinera's 2.4T (2 x 1.2T per wavelength) seventh-generation Infinite Capacity Engine (ICE7). The GX

G42 supports 100G to 1.2T line transponders and 10G to 800G client interfaces, and is suited for Communication Service Providers and Internet Content Providers, and many other network operators that require high-capacity networking with maximum spectral efficiency.

The GX G42 provides various security features and protocols including RADIUS, TACACS+, SSHv2, NTP authentication, IPsec (IKEv2), RESTCONF, NETCONF, HTTPS and TLSv1.2.

1.4 TOE Description

The Infinera GX G42 is a next-generation compact modular transport optical network platform deployed as part of a point-to-point, point-to-multipoint, or in conjunction with a FlexILS™ network for terrestrial and/or subsea applications. The GX G42 consists of one G42 chassis and provides multi-service client access (e.g., Ethernet, Optical Transport Network (OTN), etc.) to the Dense Wavelength Division Multiplexing (DWDM) transport bandwidth.

1.4.1 Clarification of Scope

Although the GX G42 performs many networking and cryptographic functions, this evaluation only addresses the functions and cryptographic libraries that provide for the security of the TOE itself as summarized in Section 1.4.1.2 and further specified in Section 5 and 6 of this Security Target. These security functions are drawn from the collaborative Protection Profile for Network Devices, Version 2.2e, 23 March 2020 (NDcPP22e).

The following functions are outside the scope of this evaluation and were not tested:

- Multi-chassis management
- Optical network communication
- USB ports (disabled in FIPS mode)
- IPv6
- Dial-out server

The following cryptographic libraries are excluded and have not been subject to evaluation. The functions they provide are outside the scope of this evaluation.

- The CHM6 module's OpenSSL library
- The CHM6 module's implementation of mbedTLS
- The ASIC Atlantic's hardware implementation of AES-GCM-256

For further details on the excluded cryptographic libraries as well as the cryptographic libraries that are included in the evaluation, see section 1.4.2 below.

1.4.2 TOE Architecture

The TOE's product type is a Network Device as defined by the NDcPP22e. The TOE is comprised of both software and hardware. The hardware is comprised of the GX G42 chassis. The software is comprised of the Converged OS (COS) version 6.2.10.

The Infinera GX G42 is a 3 RU chassis that accommodates the pluggable modules identified in the table below:

Hardware Model	Specifications
GX G42 3 RU chassis; includes 1 GX-FAN-CTRL, 2 GX-FAN-XMM4s, and 5 GX-FANMODULE-Ds: -Power Entry Module (PEM) -G42 FIPS Input/Output (I/O) Module -Fan Controller Card (FCC) Module	**Software:** Converged OS (COS) version 6.2.10 **Multi-Processor System:** XMM4 - Intel Atom C3558 (Denverton) CHM6 – NXPLS1012ASE7KKB (ARM Cortex A53)

Hardware Model	Specifications
-Fan Modules (XMM4 and sled fan) -G42 Management Control Module (XMM4) -Coherent Module ICE6 (CHM6) Transponder Module -Blank Circuit Packs -Tributary Optical Module (TOM); 400G and 100G variants (a field-replaceable, pluggable module that converts the client optical signals to and from a serial electrical signal)[1]	**ASIC**: Atlantic ASIC **Power Supply:** Power Entry Modules (PEMs)—reside in the PEM module slots (labeled as PEM 1, PEM 2, PEM 3, and PEM 4) located at the rear of the chassis (in the upper 1 RU section); up to four AC or four DC PEMs are installed to provide 12V DC power supply throughout the GX G42 **Interfaces:** 10Gbps data communication network (DCN) Ethernet RJ-45 interface. Supports 100M, 1G, and 10G Ethernet port speeds and provides a port for debug and remote management. One 10Gbps auxiliary (AUX) Ethernet interface. Supports 100M, 1G, and 10G port speeds. Two 10Gbps nodal control and timing (NCT) Ethernet interfaces supporting 100M, 1G and 10G port speeds (used for multiple chassis and thus not supported or tested in the evaluated configuration) One 1000Mbps auxiliary Ethernet interface supporting 10M, 100M and 1G port speeds. One 1000MBps craft Ethernet interface supporting 10M, 100M and 1G port speeds One console RS-232 serial port interface One Universal Serial Bus (USB) port interface[2]

Table 1 GX G42 3 RU Chassis

The GX G42 is a multi-processor system that includes multiple cryptographic libraries for cryptographic services associated with the following protocols: IPsec, IKEv2, SSHv2 and TLSv1.2.

- The XMM4 module contains the following cryptographic libraries which provide the TOE's cryptographic services in the evaluated configuration:
  - Intel OpenSSL, Strongswan and SNMP Crypto Libraries (Openssl with libSNMP and libIKE) Version 6.2 firmware. This OpenSSL library is used for IKEv2 key generation and negotiation as well as SSH and TLS cryptographic services including secret negotiation, authentication, key exchange, encryption/decryption, message authentication, hashing and DRBG.
  - Intel Kernel IPsec Crypto Library Version 4.19.274 firmware. This Kernel IPsec library is used for IPsec encryption/decryption, message authentication and hashing.
- The CHM6 module contains an OpenSSL library which provides encryption key management functions for the Atlantic ASIC and an implementation of mbedTLS used for SecureBoot. These functions are outside

[1] Optical network communication is not in the scope of this evaluation
[2] USB ports are disabled in FIPS mode

the scope of the evaluation. The CHM6 module's OpenSSL library and mbedTLS implementation are not used for any of the cryptographic functions provided by the TOE and have not been subject to evaluation.

- The ASIC Atlantic provides a hardware implementation of AES-GCM-256 that is used to provide protection from interception and integrity of user traffic. These functions are outside the scope of this evaluation. The AES-GCM-256 hardware implementation is not used for any of the cryptographic functions provided by the TOE and has not been subject to evaluation.

The GX G42 supports a next-generation converged network operating system, the Converged OS (COS) version 6.2.10, based on an open-source Linux operating system. COS offers various management interfaces and protocols, including a CLI and Web UI, as well as NETCONF and RESTCONF interfaces for flexible network monitoring and configuration.

In the evaluated configuration, the TOE supports the following management interfaces, which include a local console, and security protocols for remote management of TOE security functions.

Port No	Service	Interface	Purpose
Serial port (physical port on device)	Local console	CLI	Local console using a direct physical connection to the device for CLI access.
Transport Protocol: TCP			
830	SSH	NETCONF	Used to establish secure communication between the TOE (NETCONF server) and the NETCONF client on an external management station
8181	HTTPS	RESTCONF	Used to establish secure communication between the TOE (RESTCONF server) the RESTCONF client on an external management station
443	HTTPS	Web GUI	Used to establish secure communication between the TOE (Web GUI) and a web browser on an external management station in the operating environment.
22	SSH	CLI	Port used for CLI connection.

Table 2 Management Interfaces and Protocols

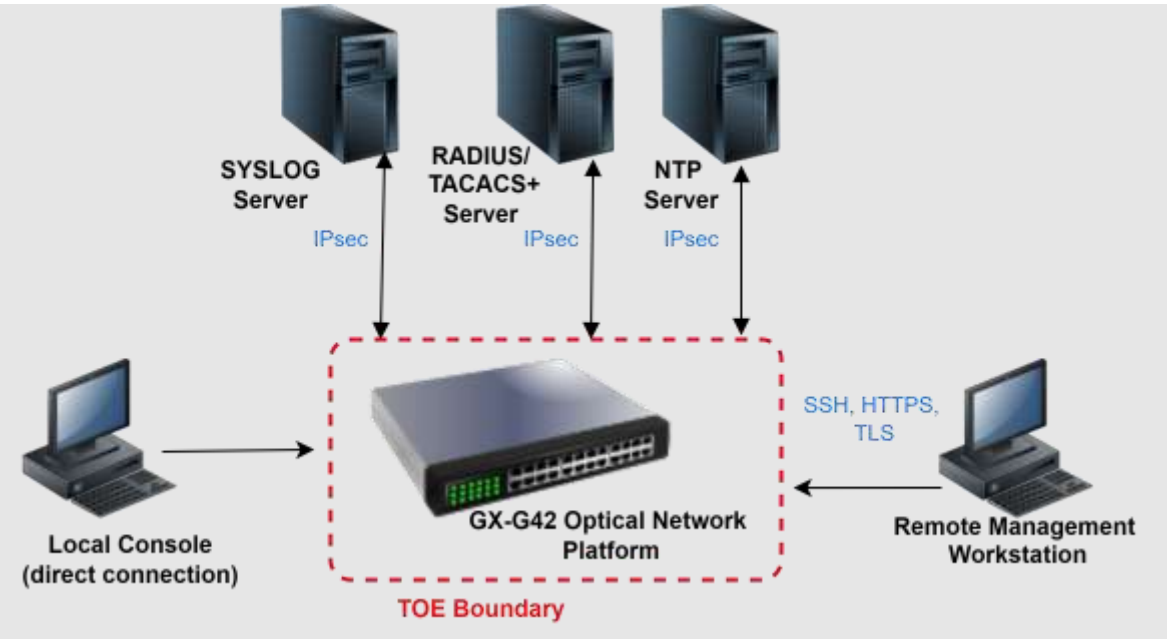
The TOE supports a number of security and access management features that offer protection against data and network tampering, including:

- Authentication, authorization, and accounting (AAA) security mechanisms, including external user authentication using a RADIUS or TACACS+ server
- User-configurable critical security parameters (CSPs), including passwords, secret keys, public key, and encryption keys
- Digitally signed images, providing integrity and authenticity of COS software during software downloads and system boot-up
- Secure protocols, including SSH, HTTPS and TLS for secure remote management and IPsec for secure communication between the TOE and the following external IT entities: a remote syslog server, remote authentication servers (RADIUS, TACACS+) and an NTP server.
- X.509 certificate management
- NTP over IPsec to securely update system time and prevent tampering of timestamps logged by the TOE

1.4.2.1 Physical Boundaries

The TOE is a single configuration whose physical boundary includes the entire chassis as identified in Table 1 above. The TOE runs Converged OS (COS) software version 6.2.10.

Error! Reference source not found. **Error! Reference source not found.** depicts a typical TOE deployment with a single instance of the TOE.



Error! Reference source not found. TOE Deployment

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

- Syslog (audit) Server – The TOE utilizes an external syslog server over a secure IPsec connection to store audit records.
- RADIUS (authentication) Server – The TOE has the ability to use RADIUS servers over a secure IPsec connection to authenticate users.
- TACACS+ (authentication) Server – The TOE has the ability to use TACACS+ servers over a secure IPsec connection to authenticate users.
- NTP (time) Server – The TOE uses a Network Time Protocol (NTP) server over a secure IPsec connection to synchronize its system clock with a central time source.
- Remote Management Workstation
  - SSH Client – The remote administrator uses an SSH client to securely access the CLI and the NETCONF API.
  - Web browser – The remote administrator uses a web browser with HTTPS/TLS to access the Web UI and the RESTCONF API.
- Local Console – The local administrator uses a direct physical connection to the TOE device.

1.4.2.2 Logical Boundaries

This section summarizes the security functions provided by Infinera GX G42 Optical Network Switch running COS version 6.2.10:

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

1.4.2.2.1 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 5 Auditable Events. The TOE stores audit logs locally so they can be accessed by an administrator and is also configured to send the logs to a designated syslog server in the operational environment.

1.4.2.2.2 Cryptographic support

The TOE includes cryptographic modules that provide key and certificate 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, HTTPS and TLS.

1.4.2.2.3 Identification and authentication

The TOE requires all administrators to be successfully identified and authenticated before allowing any TSF mediated actions to be performed except for acknowledging the pre-login warning banner. Authentication can be either locally or remotely through an external RADIUS or TACACS+ authentication server. After an administrator-specified number of failed attempts, the user account is locked out for an administrator specified period of time. 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/IKE and TLS connections.

1.4.2.2.4 Security management

The TOE provides the Security 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 Security Administrator role in accordance with FMT_SMR.2 (see Section 5.1.4.5). The TOE's Security Administrator role is further defined in Section 6.4.5. The role must have the appropriate access privileges or access will be denied.

1.4.2.2.5 Protection of the TSF

The TOE provides reliable time stamps using its own internal hardware clock or by synchronizing with an NTP server over a secure IPsec connection. The TOE stores passwords hashed in the database using PBKDF hmac-sha512. The TOE does not provide any interfaces that allow passwords or keys to be read.

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

1.4.2.2.6 TOE access

The TOE allows administrators to configure a period of inactivity for local and remote administrator 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. An administrator configured login banner is displayed at the management interfaces, local serial console (CLI), SSH CLI and Web UI (HTTPS/TLS), as an advisory and consent warning message regarding use of the TOE.

1.4.2.2.7 Trusted path/channels

The TOE uses IPsec to provide an encrypted channel between itself and the following third-party trusted IT entities in the operating environment: external syslog server, external authentication servers (RADIUS, TACACS+) and NTP server.

The TOE secures remote communication with administrators by implementing SSHv2 (CLI, NETCONF) and HTTPS/TLSv1.2 (Web UI, RESTCONF). 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.

1.4.3 TOE Documentation

The TOE includes the following guidance documents:

- Infinera GX-G42 Optical Network Platform Release 6.2.10 Hardening Guide, Revision V002, January 2025

2. Conformance Claims

This TOE is conformant to the following CC specifications:

- Common Criteria for Information Technology Security Evaluation Part 2: Security functional components, Version 3.1, Revision 5, April 2017.
    - Part 2 Extended
- Common Criteria for Information Technology Security Evaluation Part 3: Security assurance components, Version 3.1, Revision 5, April 2017.
    - Part 3 Conformant
- Package Claims:
    - collaborative Protection Profile for Network Devices, Version 2.2e, 23 March 2020 (NDcPP22e)

Package	Technical Decision	Applied	Notes
CPP_ND_V2.2E	TD0800 - Updated NIT Technical Decision for IPsec IKE/SA Lifetimes Tolerance	Yes	
CPP_ND_V2.2E	TD0792- NIT Technical Decision: FIA_PMG_EXT.1 - TSS EA not in line with SFR	Yes	
CPP_ND_V2.2E	TD0790- NIT Technical Decision: Clarification Required for testing IPv6	No	(D)TLSC is not claimed.
CPP_ND_V2.2E	TD0738 - NIT Technical Decision for Link to Allowed-With List	Yes	
CPP_ND_V2.2E	TD0670 - NIT Technical Decision for Mutual and Non-Mutual Auth TLSC Testing	No	TLSC is not claimed
CPP_ND_V2.2E	TD0639 - NIT Technical Decision for Clarification for NTP MAC Keys	Yes	
CPP_ND_V2.2E	TD0638 - NIT Technical Decision for Key Pair Generation for Authentication	Yes	
CPP_ND_V2.2E	TD0636 - NIT Technical Decision for Clarification of Public Key User Authentication for SSH	No	SSHC is not claimed
CPP_ND_V2.2E	TD0635 - NIT Technical Decision for TLS Server and Key Agreement Parameters	Yes	
CPP_ND_V2.2E	TD0632 - NIT Technical Decision for Consistency with Time Data for vNDs	Yes	
CPP_ND_V2.2E	TD0631 - NIT Technical Decision for Clarification of public key authentication for SSH Server	Yes	
CPP_ND_V2.2E	TD0592 - NIT Technical Decision for Local Storage of Audit Records	Yes	
CPP_ND_V2.2E	TD0591 - NIT Technical Decision for Virtual TOEs and hypervisors	Yes	Applied to Assumption definition
CPP_ND_V2.2E	TD0581 - NIT Technical Decision for Elliptic curve-based key establishment and NIST SP 800-56Arev3	Yes	
CPP_ND_V2.2E	TD0580 - NIT Technical Decision for clarification about use of DH14 in NDcPPv2.2e	Yes	

CPP_ND_V2.2E	TD0572 - NiT Technical Decision for Restricting FTP_ITC.1 to only IP address identifiers	Yes	
CPP_ND_V2.2E	TD0571 - NiT Technical Decision for Guidance on how to handle FIA_AFL.1	Yes	
CPP_ND_V2.2E	TD0570 - NiT Technical Decision for Clarification about FIA_AFL.1	Yes	
CPP_ND_V2.2E	TD0569 - NIT Technical Decision for Session ID Usage Conflict in FCS_DTLSS_EXT.1.7 and FCS_TLSS_EXT.1.4	Yes	
CPP_ND_V2.2E	TD0564 - NiT Technical Decision for Vulnerability Analysis Search Criteria	Yes	
CPP_ND_V2.2E	TD0563 - NiT Technical Decision for Clarification of audit date information	Yes	
CPP_ND_V2.2E	TD0556 - NIT Technical Decision for RFC 5077 question	Yes	
CPP_ND_V2.2E	TD0555 - NIT Technical Decision for RFC Reference incorrect in TLSS Test	Yes	
CPP_ND_V2.2E	TD0547 - NIT Technical Decision for Clarification on developer disclosure of AVA_VAN	Yes	
CPP_ND_V2.2E	TD0546 - NIT Technical Decision for DTLS - clarification of Application Note 63	No	DTLS(C) not claimed
CPP_ND_V2.2E	TD0537 - NIT Technical Decision for Incorrect reference to FCS_TLSC_EXT.2.3	No	TLSC not claimed
CPP_ND_V2.2E	TD0536 - NIT Technical Decision for Update Verification Inconsistency	Yes	
CPP_ND_V2.2E	TD0528 - NIT Technical Decision for Missing EAs for FCS_NTP_EXT.1.4	Yes	
CPP_ND_V2.2E	TD0527 - Updates to Certificate Revocation Testing (FIA_X509_EXT.1)	Yes	

Table 3 Technical Decisions

2.1 Conformance Rationale

The ST conforms to the NDcPP22e. The security problem definition, security objectives, and security requirements have been drawn from the PP.

3. Security Objectives

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

In general, the NDcPP22e has defined Security Objectives appropriate for network devices and as such are applicable to the Infinera GX G42 Optical Network 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.NO_GENERAL_PURPOSE There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE. Note: For vNDs the TOE includes only the contents of the its own VM, and does not include other VMs or the VS.

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

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

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

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

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

4. Extended Components Definition

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

Extended SFRs:

- NDcPP22e:FAU_STG_EXT.1: Protected Audit Event Storage
- NDcPP22e:FCS_HTTPS_EXT.1: HTTPS Protocol
- NDcPP22e:FCS_IPSEC_EXT.1: IPsec Protocol - per TD0633
- NDcPP22e:FCS_NTP_EXT.1: NTP Protocol
- NDcPP22e:FCS_RBG_EXT.1: Random Bit Generation
- NDcPP22e:FCS_SSHS_EXT.1: SSH Server Protocol - per TD0631
- NDcPP22e:FCS_TLSS_EXT.1: TLS Server Protocol Without Mutual Authentication - per TD0635
- NDcPP22e:FIA_PMG_EXT.1: Password Management
- NDcPP22e:FIA_UAU_EXT.2: Password-based Authentication Mechanism
- NDcPP22e:FIA_UIA_EXT.1: User Identification and Authentication
- NDcPP22e:FIA_X509_EXT.1/Rev: X.509 Certificate Validation
- NDcPP22e:FIA_X509_EXT.2: X.509 Certificate Authentication
- NDcPP22e:FIA_X509_EXT.3: X.509 Certificate Requests
- NDcPP22e:FPT_APW_EXT.1: Protection of Administrator Passwords
- NDcPP22e:FPT_SKP_EXT.1: Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)
- NDcPP22e:FPT_STM_EXT.1: Reliable Time Stamps - per TD0632
- NDcPP22e:FPT_TST_EXT.1: TSF testing
- NDcPP22e:FPT_TUD_EXT.1: Trusted update
- NDcPP22e: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 NDcPP22e. The refinements and operations already performed in the NDcPP22e are not identified (e.g., highlighted) here, rather the requirements have been copied from the NDcPP22e and any residual operations have been completed herein. Of particular note, the NDcPP22e made a number of refinements and completed some of the SFR operations defined in the Common Criteria (CC) and that PP should be consulted to identify those changes if necessary.

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

5.1 TOE Security Functional Requirements

The following table identifies the SFRs that are satisfied by Infinera GX G42 Optical Network Platform TOE.

Requirement Class	Requirement Component
FAU: Security audit	NDcPP22e:FAU_GEN.1: Audit Data Generation
	NDcPP22e:FAU_GEN.2: User identity association
	NDcPP22e:FAU_STG_EXT.1: Protected Audit Event Storage
FCS: Cryptographic support	NDcPP22e:FCS_CKM.1: Cryptographic Key Generation
	NDcPP22e:FCS_CKM.2: Cryptographic Key Establishment
	NDcPP22e:FCS_CKM.4: Cryptographic Key Destruction
	NDcPP22e:FCS_COP.1/DataEncryption: Cryptographic Operation (AES Data Encryption/Decryption)
	NDcPP22e:FCS_COP.1/Hash: Cryptographic Operation (Hash Algorithm)
	NDcPP22e:FCS_COP.1/KeyedHash: Cryptographic Operation (Keyed Hash Algorithm)
	NDcPP22e:FCS_COP.1/SigGen: Cryptographic Operation (Signature Generation and Verification)
	NDcPP22e:FCS_HTTPS_EXT.1: HTTPS Protocol
	NDcPP22e:FCS_IPSEC_EXT.1: IPsec Protocol - per TD0633
	NDcPP22e:FCS_NTP_EXT.1: NTP Protocol
	NDcPP22e:FCS_RBG_EXT.1: Random Bit Generation
	NDcPP22e:FCS_SSHS_EXT.1: SSH Server Protocol - per TD0631
	NDcPP22e:FCS_TLSS_EXT.1: TLS Server Protocol Without Mutual Authentication - per TD0635
FIA: Identification and authentication	NDcPP22e:FIA_AFL.1: Authentication Failure Management
	NDcPP22e:FIA_PMG_EXT.1: Password Management
	NDcPP22e:FIA_UAU.7: Protected Authentication Feedback
	NDcPP22e:FIA_UAU_EXT.2: Password-based Authentication Mechanism
	NDcPP22e:FIA_UIA_EXT.1: User Identification and Authentication
	NDcPP22e:FIA_X509_EXT.1/Rev: X.509 Certificate Validation
	NDcPP22e:FIA_X509_EXT.2: X.509 Certificate Authentication
	NDcPP22e:FIA_X509_EXT.3: X.509 Certificate Requests

FMT: Security management	NDcPP22e:FMT_MOF.1/ManualUpdate: Management of security functions behaviour
	NDcPP22e:FMT_MTD.1/CoreData: Management of TSF Data
	NDcPP22e:FMT_MTD.1/CryptoKeys: Management of TSF Data
	NDcPP22e:FMT_SMF.1: Specification of Management Functions - per TD0631
	NDcPP22e:FMT_SMR.2: Restrictions on Security Roles
FPT: Protection of the TSF	NDcPP22e:FPT_APW_EXT.1: Protection of Administrator Passwords
	NDcPP22e:FPT_SKP_EXT.1: Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)
	NDcPP22e:FPT_STM_EXT.1: Reliable Time Stamps - per TD0632
	NDcPP22e:FPT_TST_EXT.1: TSF testing
	NDcPP22e:FPT_TUD_EXT.1: Trusted update
FTA: TOE access	NDcPP22e:FTA_SSL.3: TSF-initiated Termination
	NDcPP22e:FTA_SSL.4: User-initiated Termination
	NDcPP22e:FTA_SSL_EXT.1: TSF-initiated Session Locking
	NDcPP22e:FTA_TAB.1: Default TOE Access Banners
FTP: Trusted path/channels	NDcPP22e:FTP_ITC.1: Inter-TSF trusted channel - per TD0639
	NDcPP22e:FTP_TRP.1/Admin: Trusted Path - per TD0639

Table 4 TOE Security Functional Components

5.1.1 Security audit (FAU)

5.1.1.1 Audit Data Generation (NDcPP22e:FAU_GEN.1)

NDcPP22e:FAU_GEN.1.1

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

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

b) All auditable events for the not specified level of audit; and

c) All administrative actions comprising:

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

d) Specifically defined auditable events listed in Table 5.

NDcPP22e:FAU_GEN.1.2

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

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

b) For each audit event type, based on the auditable event definitions of the functional components included in the cPP/ST, information specified in column three of Table 5.

ST Application Note: The audits generated by the start-up and shut-down of the audit functions as specified in FAU_GEN.1.1a should also include the information specified by FAU_GEN.1.2.

Requirement	Audit Event	Additional Contents
NDcPP22e:FAU_GEN.1		
NDcPP22e:FAU_GEN.2		
NDcPP22e:FAU_STG_EXT.1		
NDcPP22e:FCS_CKM.1		
NDcPP22e:FCS_CKM.2		
NDcPP22e:FCS_CKM.4		
NDcPP22e:FCS_COP.1/DataEncryption		
NDcPP22e:FCS_COP.1/Hash		
NDcPP22e:FCS_COP.1/KeyedHash		
NDcPP22e:FCS_COP.1/SigGen		
NDcPP22e:FCS_HTTPS_EXT.1	Failure to establish a HTTPS Session.	Reason for failure.
NDcPP22e:FCS_IPSEC_EXT.1	Failure to establish an IPsec SA.	Reason for failure.
NDcPP22e:FCS_NTP_EXT.1	Configuration of a new time server Removal of configured time server	Identity if new/removed time server
NDcPP22e:FCS_RBG_EXT.1		
NDcPP22e:FCS_SSHS_EXT.1	Failure to establish an SSH session.	Reason for failure.
NDcPP22e:FCS_TLSS_EXT.1	Failure to establish a TLS Session	Reason for failure
NDcPP22e:FIA_AFL.1	Unsuccessful login attempt limit is met or exceeded.	Origin of the attempt (e.g., IP address).
NDcPP22e:FIA_PMG_EXT.1		
NDcPP22e:FIA_UAU.7		
NDcPP22e:FIA_UAU_EXT.2	All use of identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
NDcPP22e:FIA_UIA_EXT.1	All use of identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
NDcPP22e:FIA_X509_EXT.1/Rev	Unsuccessful attempt to validate a certificate. Any addition, replacement or removal of trust anchors in the TOE's trust store	Reason for failure of certificate validation Identification of certificates added, replaced or removed as trust anchor in the TOE's trust store
NDcPP22e:FIA_X509_EXT.2		
NDcPP22e:FIA_X509_EXT.3		
NDcPP22e:FMT_MOF.1/ManualUpdate	Any attempt to initiate a manual update.	
NDcPP22e:FMT_MTD.1/CoreData		
NDcPP22e:FMT_MTD.1/CryptoKeys		
NDcPP22e:FMT_SMF.1	All management activities of TSF data.	
NDcPP22e:FMT_SMR.2		
NDcPP22e:FPT_APW_EXT.1		
NDcPP22e:FPT_SKP_EXT.1		
NDcPP22e:FPT_STM_EXT.1	Discontinuous changes to time - either Administrator actuated or changed via an automated process. (Note that no continuous changes to time need to be	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).

	logged. See also application note on FPT_STM_EXT.1)	
NDcPP22e:FPT_TST_EXT.1		
NDcPP22e:FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure).	
NDcPP22e:FTA_SSL.3	The termination of a remote session by the session locking mechanism.	
NDcPP22e:FTA_SSL.4	The termination of an interactive session.	
NDcPP22e:FTA_SSL_EXT.1	(if 'lock the session' is selected) Any attempts at unlocking of an interactive session. (if 'terminate the session' is selected) The termination of a local session by the session locking mechanism.	
NDcPP22e:FTA_TAB.1		
NDcPP22e:FTP_ITC.1	Initiation of the trusted channel. Termination of the trusted channel. Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempt.
NDcPP22e:FTP_TRP.1/Admin	Initiation of the trusted path. Termination of the trusted path. Failure of the trusted path functions.	

Table 5 Auditable Events

5.1.1.2 User identity association (NDcPP22e:FAU_GEN.2)

NDcPP22e:FAU_GEN.2.1
For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.

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

NDcPP22e:FAU_STG_EXT.1.1
The TSF shall be able to transmit the generated audit data to an external IT entity using a trusted channel according to FTP_ITC.1.

NDcPP22e:FAU_STG_EXT.1.2
The TSF shall be able to store generated audit data on the TOE itself. In addition
[***The TOE shall consist of a single standalone component that stores audit data locally***]

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

5.1.2 Cryptographic support (FCS)

5.1.2.1 Cryptographic Key Generation (NDcPP22e:FCS_CKM.1)

NDcPP22e:FCS_CKM.1.1

The TSF shall generate asymmetric cryptographic keys in accordance with a specified cryptographic key generation algorithm: [
- 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 'safe-prime' groups that meet the following: 'NIST Special Publication 800-56A Revision 3, Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography' and [RFC 3526, RFC 7919]].

5.1.2.2 Cryptographic Key Establishment (NDcPP22e:FCS_CKM.2)

NDcPP22e:FCS_CKM.2.1

The TSF shall perform cryptographic key establishment in accordance with a specified cryptographic key establishment method: [
- Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 3, 'Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography' (TD0581 applied),

- FFC Schemes using 'safe-prime' groups that meet the following: 'NIST Special Publication 800-56A Revision 3, 'Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography' and [groups listed in RFC 3526, RFC 7919] (TD0580 applied)].

5.1.2.3 Cryptographic Key Destruction (NDcPP22e:FCS_CKM.4)

NDcPP22e:FCS_CKM.4.1

The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method

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

that meets the following: No Standard.

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

NDcPP22e:FCS_COP.1.1/DataEncryption

The TSF shall perform encryption/decryption in accordance with a specified cryptographic algorithm AES used in [***CBC, CTR, GCM***] mode and cryptographic key sizes [***128 bits, 192 bits (GCM only), 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) (NDcPP22e:FCS_COP.1/Hash)

NDcPP22e:FCS_COP.1.1/Hash

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

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

NDcPP22e:FCS_COP.1.1/KeyedHash

The TSF shall perform keyed-hash message authentication in accordance with a specified cryptographic algorithm [***HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512***] and cryptographic key sizes [**160 bits, 256 bits, 384 bits, 512 bits**] 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) (NDcPP22e:FCS_COP.1/SigGen)

NDcPP22e:FCS_COP.1.1/SigGen

The TSF shall perform cryptographic signature services (generation and verification) in accordance with a specified cryptographic algorithm [
- RSA Digital Signature Algorithm and cryptographic key sizes (modulus) [2048 bits, 3072 bits, 4096 bits],

- Elliptic Curve Digital Signature Algorithm and cryptographic key sizes [256 bits, 384 bits, 521 bits]]
that meet the following:
[***- For RSA schemes: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSA-PKCS1v1_5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3,***

- For ECDSA schemes: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Section 6 and Appendix D, Implementing 'NIST curves' [P-256, P-384, P-521]; ISO/IEC 14888-3, Section 6.4].

5.1.2.8 HTTPS Protocol (NDcPP22e:FCS_HTTPS_EXT.1)

NDcPP22e:FCS_HTTPS_EXT.1.1

The TSF shall implement the HTTPS protocol that complies with RFC 2818.

NDcPP22e:FCS_HTTPS_EXT.1.2

The TSF shall implement HTTPS using TLS.

NDcPP22e:FCS_HTTPS_EXT.1.3

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

5.1.2.9 IPsec Protocol - per TD0633 (NDcPP22e:FCS_IPSEC_EXT.1)

NDcPP22e:FCS_IPSEC_EXT.1.1

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

NDcPP22e:FCS_IPSEC_EXT.1.2

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

NDcPP22e:FCS_IPSEC_EXT.1.3

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

NDcPP22e:FCS_IPSEC_EXT.1.4

The TSF shall implement the IPsec protocol ESP as defined by RFC 4303 using the cryptographic

algorithms [***AES-GCM-128 (RFC 4106), AES-GCM-192 (RFC 4106), AES-GCM-256 (RFC 4106)***] together with a Secure Hash Algorithm (SHA)-based HMAC [***HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512***].

NDcPP22e: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]].

NDcPP22e:FCS_IPSEC_EXT.1.6

The TSF shall ensure the encrypted payload in the [***IKEv2***] protocol uses the cryptographic algorithms [***AES-GCM-128, AES-GCM-192, AES-GCM-256 (specified in RFC 5282)***].

NDcPP22e:FCS_IPSEC_EXT.1.7

The TSF shall ensure that [

- IKEv2 SA lifetimes can be configured by a Security Administrator based on [o length of time, where the time values can be configured within [1 to 24] hours]].

NDcPP22e:FCS_IPSEC_EXT.1.8

The TSF shall ensure that [

- IKEv2 Child SA lifetimes can be configured by a Security Administrator based on [

o number of bytes,

o length of time, where the time values can be configured within [1 to 8] hours]].

NDcPP22e:FCS_IPSEC_EXT.1.9

The TSF shall generate the secret value x used in the IKE Diffie-Hellman key exchange ('x' in g^x mod p) using the random bit generator specified in FCS_RBG_EXT.1, and having a length of at least [***224 bits (for DH Group 14), 256 bits (for DH Group 15), 320 bits (for DH Group 16), 384 bits (for DH Group 17), 512 bits for DH Group 18), 256 bits (for DH Group 19), 384 bits (for DH Group 20) and 521 bits (for DH Group 21)***] bits.

NDcPP22e: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].

NDcPP22e:FCS_IPSEC_EXT.1.11

The TSF shall ensure that IKE protocols implement DH Group(s) [***[14 (2048-bit MODP), 15 (3072-bit MODP), 16 (4096-bit MODP), 17 (6144-bit MODP), 18 (8192-bit MODP)] according to RFC 3526], [19 (256-bit Random ECP), 20 (384-bit Random ECP), 21 (521-bit Random ECP) according to RFC 5114]***].

NDcPP22e:FCS_IPSEC_EXT.1.12

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

NDcPP22e:FCS_IPSEC_EXT.1.13

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

NDcPP22e:FCS_IPSEC_EXT.1.14

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

5.1.2.10 NTP Protocol (NDcPP22e:FCS_NTP_EXT.1)

NDcPP22e:FCS_NTP_EXT.1.1

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

NDcPP22e:FCS_NTP_EXT.1.2

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

NDcPP22e:FCS_NTP_EXT.1.3
The TSF shall not update NTP timestamp from broadcast and/or multicast addresses.
NDcPP22e:FCS_NTP_EXT.1.4
The TSF shall support configuration of at least three (3) NTP time sources in the Operational Environment.

5.1.2.11 Random Bit Generation (NDcPP22e:FCS_RBG_EXT.1)

NDcPP22e:FCS_RBG_EXT.1.1
The TSF shall perform all deterministic random bit generation services in accordance with ISO/IEC 18031:2011 using [***CTR_DRBG (AES)***].
NDcPP22e:FCS_RBG_EXT.1.2
The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [***[1] platform-based noise source***] with a minimum of [256 ***bits***] of entropy at least equal to the greatest security strength, according to ISO/IEC 18031:2011Table C.1 'Security Strength Table for Hash Functions', of the keys and hashes that it will generate.

5.1.2.12 SSH Server Protocol - per TD0631 (NDcPP22e:FCS_SSHS_EXT.1)

NDcPP22e:FCS_SSHS_EXT.1.1
The TSF shall implement the SSH protocol that complies with: RFC(s) 4251, 4252, 4253, 4254, [***4256, 5647, 5656, 8308 section 3.1, 8332***].
NDcPP22e:FCS_SSHS_EXT.1.2
The TSF shall ensure that the SSH protocol implementation supports the following user authentication methods as described in RFC 4252: public key-based, [***password-based***]. (TD0631 applied)
NDcPP22e:FCS_SSHS_EXT.1.3
The TSF shall ensure that, as described in RFC 4253, packets greater than [262130] bytes in an SSH transport connection are dropped.
NDcPP22e: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, aes128-gcm@openssh.com, aes256-gcm@openssh.com***].
NDcPP22e:FCS_SSHS_EXT.1.5
The TSF shall ensure that the SSH public-key based authentication implementation uses [***rsa-sha2-256, rsa-sha2-512, ecdsa-sha2-nistp256, ecdsa-sha2-nistp384, ecdsa-sha2-nistp521***] as its public key algorithm(s) and rejects all other public key algorithms.
NDcPP22e:FCS_SSHS_EXT.1.6
The TSF shall ensure that the SSH transport implementation uses [***hmac-sha2-256, hmac-sha2-512, implicit***] as its MAC algorithm(s) and rejects all other MAC algorithm(s).
NDcPP22e:FCS_SSHS_EXT.1.7
The TSF shall ensure that [***ecdh-sha2-nistp256***] and [***diffie-hellman-group14-sha256***, ***ecdh-sha2-nistp384, ecdh-sha2-nistp521***] are the only allowed key exchange methods used for the SSH protocol.

NDcPP22e:FCS_SSHS_EXT.1.8
The TSF shall ensure that within SSH connections, the same session keys are used for a threshold of no longer than one hour, and each encryption key is used to protect no more than one gigabyte of data. After any of the thresholds are reached, a rekey needs to be performed.

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

NDcPP22e:FCS_TLSS_EXT.1.1
The TSF shall implement [***TLS 1.2 (RFC 5246)***] and reject all other TLS and SSL versions. The TLS implementation will support the following ciphersuites:
[***TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246,***

TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246.
TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288,
TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288,
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289,
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289,
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289,
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289,
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289,
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289,
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289,
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289]
and no other ciphersuites.

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

NDcPP22e:FCS_TLSS_EXT.1.3
The TSF shall perform key establishment for TLS using [***Diffie-Hellman groups [ffdhe2048, ffdhe3072, ffdhe4096, ffdhe6144, ffdhe8192], ECDHE curves [secp256r1, secp384r1, secp521r1] and no other curves***]].

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

5.1.3 Identification and authentication (FIA)

5.1.3.1 Authentication Failure Management (NDcPP22e:FIA_AFL.1)

NDcPP22e:FIA_AFL.1.1
The TSF shall detect when an Administrator configurable positive integer within [**1 to 255**] unsuccessful authentication attempts occur related to Administrators attempting to authenticate remotely using a password.

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

5.1.3.2 Password Management (NDcPP22e:FIA_PMG_EXT.1)

NDcPP22e:FIA_PMG_EXT.1.1
The TSF shall provide the following password management capabilities for administrative passwords:
a) Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters: [*'!', '@', '#', '$', '%', '^', '&', '*', '(', ')' [and additional special characters: <sp>* ~ _ - + = ` | { } " [ ] : ; < > , . ? / ' \ *]*];
b) Minimum password length shall be configurable to between [**8**] and [**200**] characters.

5.1.3.3 Protected Authentication Feedback (NDcPP22e:FIA_UAU.7)

NDcPP22e:FIA_UAU.7.1
The TSF shall provide only obscured feedback to the administrative user while the authentication is in progress at the local console.

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

NDcPP22e:FIA_UAU_EXT.2.1
The TSF shall provide a local [***password-based, SSH public key-based [remote password based***

authentication via RADIUS, remote password based authentication via TACACS+/] authentication mechanism to perform local administrative user authentication.

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

NDcPP22e:FIA_UIA_EXT.1.1

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

NDcPP22e:FIA_UIA_EXT.1.2

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

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

NDcPP22e:FIA_X509_EXT.1.1/Rev

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

NDcPP22e:FIA_X509_EXT.1.2/Rev

The TSF shall only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.

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

NDcPP22e:FIA_X509_EXT.2.1

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

NDcPP22e:FIA_X509_EXT.2.2

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

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

NDcPP22e:FIA_X509_EXT.3.1

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

NDcPP22e:FIA_X509_EXT.3.2

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

5.1.4 Security management (FMT)

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

NDcPP22e:FMT_MOF.1.1/ManualUpdate

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

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

NDcPP22e:FMT_MTD.1.1/CoreData

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

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

NDcPP22e:FMT_MTD.1.1/CryptoKeys

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

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

NDcPP22e:FMT_SMF.1.1

The TSF shall be capable of performing the following management functions:
- Ability to administer the TOE locally and remotely;
- Ability to configure the access banner;
- Ability to configure the session inactivity time before session termination or locking;
- Ability to update the TOE, and to verify the updates using [***digital signature***] capability prior to installing those updates;
- Ability to configure the authentication failure parameters for FIA_AFL.1;
- [
Ability to modify the behavior of the transmission of audit data to an external IT entity,
Ability to configure the list of TOE-provided services available before an entity is identified and authenticated, as specified in FIA_UIA_EXT.1,
Ability to manage the cryptographic keys,
Ability to configure the cryptographic functionality,
Ability to configure the lifetime for IPsec SAs,
Ability to set the time which is used for time-stamps;
Ability to configure NTP,
Ability to configure the reference identifier for the peer;
Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors,
Ability to import X509v3 certificates to the TOE's trust store,
Ability to manage the trusted public keys database].
(TD0631 applied)

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

NDcPP22e:FMT_SMR.2.1

The TSF shall maintain the roles: - Security Administrator.

NDcPP22e:FMT_SMR.2.2

The TSF shall be able to associate users with roles.

NDcPP22e:FMT_SMR.2.3

The TSF shall ensure that the conditions
- The Security Administrator role shall be able to administer the TOE locally;

- The Security Administrator role shall be able to administer the TOE remotely
are satisfied.

5.1.5 Protection of the TSF (FPT)

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

NDcPP22e:FPT_APW_EXT.1.1
The TSF shall store administrative passwords in non-plaintext form.
NDcPP22e:FPT_APW_EXT.1.2
The TSF shall prevent the reading of plaintext administrative passwords.

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

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

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

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

5.1.5.4 TSF testing (NDcPP22e:FPT_TST_EXT.1)

NDcPP22e:FPT_TST_EXT.1.1
The TSF shall run a suite of the following self-tests [***during initial start-up (on power on), periodically during normal operation, at the request of the authorised user***] to demonstrate the correct operation of the TSF: [c**ryptographic known answer tests (KAT), pair-wise consistency tests, KDF tests and software integrity test**].

5.1.5.5 Trusted update (NDcPP22e:FPT_TUD_EXT.1)

NDcPP22e:FPT_TUD_EXT.1.1
The TSF shall provide Security Administrators the ability to query the currently executing version of the TOE firmware/software and [***no other TOE firmware/software version***].
NDcPP22e:FPT_TUD_EXT.1.2
The TSF shall provide Security Administrators the ability to manually initiate updates to TOE firmware/software and [***no other update mechanism***].
NDcPP22e:FPT_TUD_EXT.1.3
The TSF shall provide means to authenticate firmware/software updates to the TOE using a [***digital signature***] prior to installing those updates.

5.1.6 TOE access (FTA)

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

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

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

NDcPP22e:FTA_SSL.4.1

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

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

NDcPP22e:FTA_SSL_EXT.1.1

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

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

NDcPP22e:FTA_TAB.1.1

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

5.1.7 Trusted path/channels (FTP)

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

NDcPP22e:FTP_ITC.1.1

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

NDcPP22e:FTP_ITC.1.2

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

NDcPP22e:FTP_ITC.1.3

The TSF shall initiate communication via the trusted channel for [

- **external syslog server using IPsec**
- **external RADIUS authentication server using IPsec**
- **external TACACS+ authentication server using IPsec**
- **external NTP server using IPsec**].

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

NDcPP22e:FTP_TRP.1.1/Admin

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

NDcPP22e:FTP_TRP.1.2/Admin

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

NDcPP22e:FTP_TRP.1.3/Admin

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

5.2 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 6 Assurance Components

5.2.1 Development (ADV)

5.2.1.1 Basic Functional Specification (ADV_FSP.1)

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

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

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

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

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

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

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

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

5.2.2 Guidance documents (AGD)

5.2.2.1 Operational User Guidance (AGD_OPE.1)

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

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

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

AGD_OPE.1.3c

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

AGD_OPE.1.4c

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

AGD_OPE.1.5c

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

AGD_OPE.1.6c

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

AGD_OPE.1.7c

The operational user guidance shall be clear and reasonable.

AGD_OPE.1.1e

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

5.2.2.2 Preparative Procedures (AGD_PRE.1)

AGD_PRE.1.1d

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

AGD_PRE.1.1c

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

AGD_PRE.1.2c

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

AGD_PRE.1.1e

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

AGD_PRE.1.2e

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

5.2.3 Life-cycle support (ALC)

5.2.3.1 Labelling of the TOE (ALC_CMC.1)

ALC_CMC.1.1d

The developer shall provide the TOE and a reference for the TOE.

ALC_CMC.1.1c

The TOE shall be labelled with its unique reference.

ALC_CMC.1.1e

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

5.2.3.2 TOE CM Coverage (ALC_CMS.1)

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

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

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

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

5.2.4 Tests (ATE)

5.2.4.1 Independent Testing - Conformance (ATE_IND.1)

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

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

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

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

5.2.5 Vulnerability assessment (AVA)

5.2.5.1 Vulnerability Survey (AVA_VAN.1)

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

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

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

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

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

6. TOE Summary Specification

This chapter describes the security functions:

- Security audit
- Cryptographic support
- Identification and authentication
- Security management
- 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 5 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 5.

All cryptographic keys are assigned an administrator specified name upon generation or import. This unique key name is included in the audit logs for the administrator actions of generating/import of, changing, or deleting of cryptographic keys.

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. If the IPsec connection fails or the remote syslog server is otherwise unavailable, the administrator may recover messages from local storage. When the connection is re-established, only new audits will be sent.

When the local audit storage is full, the TOE will overwrite the oldest audit records first using a FIFO (first in, first out) replacement mode performed via rotation of log files. The TOE retains 10 log files which are rotated when they exceed 30 MB. The log files are stored in a filesystem of 8.4GB with typically 3.0GB of available space. The TOE will issue a warning message when there is only 10% storage space remaining. The TOE will not allow the configuration of a smaller rotation size or lower number of log files to be retained.

An authorized administrator must log into the TOE in order to view the local audit records. There are no interfaces via which the administrator can clear/delete or otherwise modify the contents of the local audit logs.

6.2 Cryptographic support

The TOE is a multi-processor system that includes multiple cryptographic libraries for cryptographic services associated with the following protocols: IPsec, IKEv2, SSH and TLSv1.2.

- The XMM4 module contains the following cryptographic libraries which provide the cryptographic services in the evaluated configuration:

- Intel OpenSSL, Strongswan and SNMP Crypto Libraries (Openssl with libSNMP and libIKE) Version 6.2 firmware. This OpenSSL library is used for IKEv2 key generation and negotiation as well as SSH and TLS cryptographic services including secret negotiation, authentication, key exchange, encryption/decryption, message authentication, hashing and DRBG.
- Intel Kernel IPsec Crypto Library Version 4.19.274 firmware. This Kernel IPsec library is used for IPsec encryption/decryption, message authentication and hashing.

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:

SFR	Algorithm	NIST/ISO Standard	Certificate #	
			Intel Kernel IPsec Crypto LIbrary firmware 4.19.274	Intel Openssl with libSNMP, libIKE firmware version 6.2
FCS_CKM.1	RSA KeyGen (2048, 3072, 4096 bits)	FIPS PUB 186-4		A4958
	ECDSA KeyGen/KeyVer P-256, P-384 and P-521	FIPS PUB 186-4		A4958
	FFC Safe Prime Groups (DH-14/15/16/17/18/ 19/20/21 ffdhe2048, ffdhe3072, ffdhe4096, ffdhe6144, ffdhe8192)	NIST SP 800-56A RFC 3526 RFC 7919	Tested with known good implementation	
FCS_CKM.2	KAS ECC	NIST SP 800-56A		A4958
	FFC Safe Prime Groups (DH-14/15/16/17/18/ 19/20/21 ffdhe2048, ffdhe3072, ffdhe4096, ffdhe6144, ffdhe8192)	NIST SP 800-56A RFC 3526 RFC 7919	Tested with known good implementation	
FCS_COP.1/ DataEncryption	AES CBC (128, 192 and 256 bits)	CBC as specified in ISO 10116, GCM as specified in ISO 19772		A4958
	AES GCM (128, 192 and 256 bits)	GCM as specified in ISO 19772	A4956	A4958
	AES CTR (128 and 256 bits)	CTR as specified in ISO 10116		A4958
FCS_COP.1/SigGen	RSA SigGen/SigVer (2048 bits, 3072, 4096 bits)	FIPS PUB 186-4		A4958
	ECDSA SigGen/SigVer (P-256, P-384, P-521)	FIPS PUB 186-4		A4958
FCS_COP.1/Hash	SHA-1/256/384/512 (digest sizes 160, 256, 384, 512 bits)	ISO/IEC 10118-3:2004	A4956	A4958
FCS_COP.1/ KeyedHash	HMAC-SHA-1, HMAC-SHA-256,	ISO/IEC 9797-2:2011, Section 7	A4956	A4958

SFR	Algorithm	NIST/ISO Standard	Certificate #	
			Intel Kernel IPsec Crypto LIbrary firmware 4.19.274	Intel Openssl with libSNMP, libIKE firmware version 6.2
	HMAC-SHA-384, HMAC-SHA-512 (digest sizes 160, 256, 384, 512 bits, respectively)	'MAC Algorithm 2'		
FCS_RBG_EXT.1	CTR_DRBG (AES) with platform based noise source (256 bits)	ISO/ICE 18031:2011		A4958

Table 7 Algorithm Certificate Numbers

6.2.1 FCS_CKM.1

The TOE supports key generation using RSA schemes (2048, 3072, 4096 bits) that meet FIPS PUB 186-4, Appendix B.3 and ECC schemes (P-256, P-384, P-521) that meet FIPS PUB 186-4, Appendix B.4 for the following:

- Generating key pairs for certificate signing requests (IPsec, TLS).
- Generates ECDHE keys for key exchange/establishment (SSH, TLS)
- Signature generation and verification for authentication (IPsec, TLS, SSH)

The TOE implements FFC schemes using 'safe-prime' groups that meet NIST SP 800-56A Revision 3 for the following:

- Generates DH keys for FFC key establishment (TLS)
- Generates keys for Diffie-Hellman groups for key exchange according to both RFC 3526 and RFC 5114 (IPsec, SSH).

6.2.2 FCS_CKM.2

The following table identifies the supported key establishment schemes mapped to the associated SFRs and their usage. These key establishment schemes correspond to the key generation schemes identified in FCS_CKM.1.1.

Scheme	SFR	Services
ECC (P-256/384/521) DH-19/20/21	FCS_IPSEC_EXT.1	Syslog over IPsec RADIUS over IPsec TACACS+ over IPsec NTP over IPsec
ECC (P-256/384/521)	FCS_SSHS_EXT.1	SSH Remote Administration
	FCS_TLSS_EXT.1	HTTPS/TLS Remote Administration
FFC schemes using safe prime groups (DH-14/15/16/17/18)	FCS_IPSEC_EXT.1	IKEv2 key exchange (Syslog, RADIUS, TACACS+ and NTP over IPsec)
FFC schemes using safe prime groups (ffdhe2048, ffdhe3072, ffdhe4096, ffdhe6144, ffdhe8192)	FCS_TLSS_EXT.1	HTTPS/TLS Remote Administration
FFC schemes using safe	FCS_SSHS_EXT.1	SSH Remote Administration

Scheme	SFR	Services
prime groups (DH group 14)		

Table 8 Key Establishment Schemes

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 zeroes.

The following table presents the crypto security parameters (CSPs), secret keys, and private keys provided by the TOE. The table also identifies where each CSP is stored and when each CSP or key is cleared.

Key/CSP Name/Type	Security Function	Generation	Storage	Zeroization	Use & related keys
User Password	PBKDF2 with HMAC-SHA2-512	User generated	Ciphertext in Flash (protected by HMAC-SHA2-512)	Zeroized by SSP/CSP/PSP Zeroization Command	User Authentication.
DRBG Entropy Input	CTR_DRBG	Generated from Entropy Source	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, power cycle	Random Number Generation
DRBG Seed, Internal State V value, and Key	CTR_DRBG	Internally Derived from entropy input string as defined by SP800-90A	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, power cycle	Random Number Generation
SSHv2 DH Private Key	KAS (FFC); KAS-FFC-SSC; Safe Primes Key Generation	Internally generated conformant to SP800-133r2 (CKG) using SP800-56A rev3 Diffie-Hellman key generation method, and the random value used in key generation is generated using SP800-90A DRBG	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, power cycle or handshake completion	Used to derive SSHv2 DH Shared Secret
SSHv2 DH Public Key	KAS (FFC); KAS-FFC-SSC; Safe Primes Key Generation	Internally derived per the Diffie-Hellman key agreement (SP800-56A rev3)	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, power cycle or handshake completion	Used to derive SSHv2 DH Shared Secret
SSHv2 DH Shared Secret	KAS-FFC-SSC; KAS-FFC	Internally generated using SP800-56Ar3 DH shared secret computation	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, power cycle or handshake completion	Used to derive SSHv2 Session Encryption Key and SSHv2 Session Integrity Key

Key/CSP Name/Type	Security Function	Generation	Storage	Zeroization	Use & related keys
SSHv2 ECDH Private Key	KAS (ECC); KAS-ECC-SSC	Internally generated conformant to SP800-133r2 (CKG) using SP800-56A rev3 EC Diffie-Hellman key generation method, and the random value used in key generation is generated using SP800-90A DRBG	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, power cycle or handshake completion	Used to derive SSHv2 ECDH Shared Secret
SSHv2 ECDH Public Key	KAS (ECC); KAS-ECC77-SSC	Internally derived per the EC Diffie-Hellman key agreement (SP800-56Arev3)	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, power cycle or handshake completion	Used to derive SSHv2 ECDH Shared Secret
SSHv2 ECDH Shared Secret	KAS (ECC); KAS-ECC-SSC	Internally generated using SP800-56Ar3 ECDH shared secret computation	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, power cycle or handshake completion	Used to derive SSHv2 Session Encryption Key and SSH Session Integrity Key
SSHv2 ECDSA Host Private Key	ECDSA (KeyGen, KeyVer and SigGen)	Internally generated conformant to SP800-133r2 (CKG) using FIPS 186-4 ECDSA key generation method, and the random value used in key generation is generated using SP800-90A DRBG	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, new host key generated	Used for SSHv2 authentication
SSHv2 ECDSA Host Public Key	ECDSA (KeyGen, KeyVer and SigVer)	Internally derived per the FIPS 186-4 ECDSA key generation method	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, new host key generated	Used for SSHv2 authentication
SSHv2 RSA Host Private Key	RSA (KeyGen and SigGen)	Internally generated conformant to SP800-133r2 (CKG) using FIPS 186-4 RSA key generation method, and the random value used in the key generation is generated using SP800-90A DRBG	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, new host key generated	Used for SSHv2 authentication

Key/CSP Name/Type	Security Function	Generation	Storage	Zeroization	Use & related keys
SSHv2 RSA Host Public Key	RSA (KeyGen and SigVer)	Internally derived per the FIPS 186-4 RSA key generation method	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, new host key generated	Used for SSHv2 authentication
SSHv2 Session Encryption Key	AES-CTR AES-GCM	Internally derived via key derivation function defined in SP800-135 KDF (SSH)	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command power cycle or session termination	Used to secure SSHv2 session confidentiality
SSHv2 Session Integrity Key	hmac-sha2-256, hmac-sha2-512	Internally derived per the key derivation function defined in SP800-135 KDF SSH	Plaintext in RAM	Zeroized by Zeroization Command power cycle or session termination	Used to secure SSHv2 session integrity
IPSec/IKEv2 DH Private Key	KAS-FFC-SSC; KAS-FFC	Internally generated. conformant to SP800-133r2 (CKG) using SP800-56Arev3 Diffie-Hellman key generation method, and the random value used in key generation is generated using SP800-90A DRBG	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, power cycle or handshake completion	Used to derive IKEv2 DH Shared Secret
IPSec/IKEv2 DH Public Key	KAS-FFC-SSC, KAS-FFC	Internally derived internally per the Diffie-Hellman key agreement (SP800-56Arev3)	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, on power cycle or handshake completion	Used to derive IKEv2 DH Shared Secret
IPSec/IKEv2 DH Shared Secret	KAS-FFC-SSC, KAS-FFC	Internally derived using SP800-56A rev3 Diffie-Hellman shared secret computation	DRAM (plaintext)	Zeroized by SSP/CSP/PSP Zeroization Command, on power cycle or handshake completion	Used to derive IPSec/IKEv2 Session Encryption Keys, IPSec/IKEv2 Authentication Keys
IPSec/IKEv2 ECDH Private Key	KAS-ECC-SSC, KAS-ECC	Internally generated conformant to SP800-133r2 (CKG) using SP800-56A rev3 EC Diffie-Hellman key generation method, and the random value used in key generation is generated using SP800-90A DRBG	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, power cycle or handshake completion	Used to derive IKEv2 ECDH Shared Secret

Key/CSP Name/Type	Security Function	Generation	Storage	Zeroization	Use & related keys
IPSec/IKEv2 ECDH Public Key	KAS-ECC-SSC, KAS-ECC	Internally derived internally per the EC Diffie-Hellman key agreement (SP800-56Arev3)	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, power cycle or handshake completion	Used to derive IKEv2 ECDH Shared Secret
IPSec/IKEv2 ECDH Shared Secret	KAS-ECC-SSC, KAS-ECC	Internally derived internally per the EC Diffie-Hellman key agreement (SP800-56Arev3)	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, power cycle or handshake completion	Used to derive IKEv2 ECDH Shared Secret
IPSec/IKEv2 Authentication Private Key	RSA SigGen, ECDSA SigGen	Internally generated conformant to SP800-133r2 (CKG) using FIPS 186-4 ECDSA/RSA key generation method, and the random value used in key generation is generated using SP800-90A DRBG	Plaintext in RAM and encrypted in FLASH The key is stored in encrypted form in the database. The database is stored on flash using LUKS which uses AES-XTS-plain64 with 512-bit-keys.	Zeroized by SSP/CSP/PSP Zeroization Command,	Used for IPSec/IKEv2 authentication
IPSec/IKEv2 Authentication Public Key	RSA SigVer, ECDSA SigVer	Internally generated conformant to SP800-133r2 (CKG) using FIPS 186-4 ECDSA/RSA key generation method, and the random value used in key generation is generated using SP800-90A DRBG	Plaintext in flash	Zeroized by SSP/CSP/PSP Zeroization Command	Used for IPSec/IKEv2 authentication
IPSec/IKEv2 Pre-shared Secret	Pre-shared key for IPsec/IKE authentication	User generated	Plaintext in Flash	Zeroized by SSP/CSP/PSP Zeroization Command	Used for IPSec/IKEv2 authentication
SKEYSEED	IKE-KDF	It was derived via key derivation function defined in SP800-135rev1 KDF (IKEv2)	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, power cycle or handshake completion	Keying material used to derive the IPSec/IKEv2 Session Encryption Key and IPSec/IKEv2 Authentication Key
IPSec/IKEv2 Session Encryption Key	AES-GCM	Internally derived via key derivation function defined in SP800-135rev1 KDF (IKEv2)	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, power cycle or session termination	Used to secure IPSec/IKEv2 session confidentiality,
IPSec/IKEv2 Authentication Key	HMAC-SHA1 HMAC-SHA2-	Internally derived via key derivation	Plaintext in RAM	Zeroized by SSP/CSP/PSP	Used to secure IPSec/IKEv2 session

Key/CSP Name/Type	Security Function	Generation	Storage	Zeroization	Use & related keys
	256, HMAC-SHA2-384, HMAC-SHA2-512	function defined in SP800-135rev1 KDF (IKEv2)		Zeroization Command, power cycle or session termination	integrity
TLSv1.2 DH Private Key	KAS-FFC-SSC, KAS-FFC	Internally generated conformant to SP800-133r2 (CKG) using SP800-56A rev3 Diffie-Hellman key generation method, and the random value used in key generation is generated using SP800-90A DRBG	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, power cycle or handshake completion.	Used to derive TLSv1.2 DH Shared Secret
TLSv1.2 DH Public Key	KAS-FFC-SSC, KAS-FFC	Internally derived internally per the Diffie-Hellman key agreement (SP800-56Arev3)	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, power cycle or handshake completion	Used to derive TLSv1.2 DH Shared Secret
TLSv1.2 DH Shared Secret	KAS-ECC-SSC	Internally derived internally per the Diffie-Hellman key agreement (SP800-56Arev3)	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, power cycle or handshake completion	Used to derive TLSv1.2 Session Encryption Key and TLS Session Authentication Key
TLSv1.2 ECDH Private Key	KAS-ECC-SSC, KAS-ECC	Internally generated conformant to SP800-133r2 (CKG) using SP800-56A rev3 Diffie-Hellman key generation method, and the random value used in key generation is generated using SP800-90A DRBG	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, power cycle or handshake completion	Used to derive TLSv1.2 ECDH Shared Secret
TLSv1.2 ECDH Public Key	KAS-ECC-SSC, KAS-ECC	Internally derived internally per the Diffie-Hellman key agreement (SP800-56Arev3)	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, power cycle or handshake completion	Used to derive TLSv1.2 ECDH Shared Secret
TLSv1.2 ECDH Shared Secret	KAS-ECC-SSC, KAS-ECC	Internally derived internally per the	Plaintext in RAM	Zeroized by SSP/CSP/PSP	Used to derive TLSv1.2 Session Encryption Key

Key/CSP Name/Type	Security Function	Generation	Storage	Zeroization	Use & related keys
		Diffie-Hellman key agreement (SP800-56Arev3)		Zeroization Command, power cycle or handshake completion	and TLS Session Authentication Key
TLSv1.2 RSA Private Key	RSA SigGen,	Internally generated conformant to SP800-133r2 (CKG) using FIPS 186-4 RSA/RSA key generation method, and the random value used in key generation is generated using SP800-90A DRBG	Plaintext in flash	Zeroized by SSP/CSP/PSP Zeroization Command	Used for TLSv1.2 authentication
TLSv1.2 RSA Public Key	RSA SigVer,	Internally derived per the FIPS186-4 RSA Keypair generation method	Plaintext in flash	Zeroized by SSP/CSP/PSP Zeroization Command	Used for TLSv1.2 authentication
TLSv1.2 Master Secret	KDF-TLSv1.2,	Internally derived via key derivation function defined in SP800-135rev1 KDF (TLSv1.2)	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, power cycle or session termination	Used to derive TLSv1.2 Session Encryption Key and TLSv1.2 Session Authentication Key
TLSv1.2 Session Encryption Key	AES-CBC, AES-GCM, TLSv1.2-KDF,	Internally derived via key derivation function defined in SP800-135 rev1 KDF TLSv1.2	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, power cycle or session termination	Used to secure TLSv1.2 session confidentiality
TLSv1.2 Session Authentication Key	HMAC-SHA2-256, HMAC-SHA2-384, HMAC-SHA2-512, TLSv1.2-KDF,	Internally derived via key derivation function defined in SP800-135 rev1 KDF TLSv1.2 KDF	Plaintext in RAM	Zeroized by SSP/CSP/PSP Zeroization Command, power cycle or session termination	Used to secure TLSv1.2 session integrity
RADIUS shared secret	RADIUS shared secret (password authentication)	User Generated	Ciphertext in Flash (protected by HMAC-SHA2-512)	Zeroized by SSP/CSP/PSP Zeroization Command	Used as part of RADIUS authentication.
TACACS+ shared secret	TACACS+ shared secret (password authentication)	User Generated	Ciphertext in Flash (protected by HMAC-SHA2-512)	Zeroized by SSP/CSP/PSP Zeroization Command	Used as part of TACACS+ authentication.

Table 9 Cryptographic Security Parameters

6.2.4 FCS_COP.1/DataEncryption

The TOE supports AES GCM (128, 192 and 256 bits) for data encryption/decryption in IPSec. The TOE supports AES GCM (128 and 256 bits) and AES CTR (128 and 256 bits) for data encryption/decryption in SSH. The TOE supports AES_128_CBC, AES_256_CBC, AES_128_GCM and 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)	Protocol(s) used in
HMAC-SHA-1	160	512	160	IPsec/IKEv2
HMAC-SHA2-256	256	512	256	IPsec/IKEv2, TLS, SSH
HMAC-SHA2-384	384	1024	384	IPsec/IKEv2, TLS
HMAC-SHA2-512	512	1024	512	IPsec/IKEv2, SSH

Table 10 Keyed Hash Algorithms

6.2.7 FCS_COP.1/SigGen

The TOE supports RSA (modulus 2048, 3072 and 4096) and ECDSA with elliptical curve size P-256, P-384 and P-521 for signature generation and verification.

6.2.8 FCS_HTTPS_EXT.1

The TOE complies to RFC 2818 by implementing HTTP over TLS channels. The TOE provides an HTTPS/TLS interface for remote administration via the Web UI and RESTCONF. These interfaces do not require or support client certificate authentication. The TOE follows common practice identified in RFC 2818 by not using the standard HTTP port 80. In accordance with RFC2818 the TOE is prepared to receive an incomplete close from the client and the TOE attempts to initiate an exchange of closure alerts with the client before closing the connection.

6.2.9 FCS_IPSEC_EXT.1

The TOE implements IPsec to provide an authenticated and encrypted channel between the TOE and the following remote IT entities: a Syslog server, a RADIUS server, a TACACS+ server and an NTP server.

The TOE can be configured with SPD rules that will either Bypass (send/receive the packets without IPsec protection), Protect (secure the packet via IPsec), or Discard (drop the packets) based on IP addresses and port numbers. The TOE can also be configured with a default discard SPD rule that will discard all traffic that does not match any other SPD rule. The configuration of the TOE's SPD includes settings for the IP protocol number, local IP address and port number, remote IP address and port number, the IP processing choices or rule actions (ie. protect, bypass or discard), the mode (tunnel or transport) and the IPsec ESP cryptographic algorithms.

The SPD priority is configurable for each SPD rule. The SPD is used if the traffic selector matches the traffic and the priority is the lowest (ie. 1 has priority over 9999) in the system. On ingress traffic, interface ACLs are processed prior to IPsec. On egress traffic, interface ACLs are processed after IPsec.

The TOE implements IPsec in accordance with RFC 4301 and supports both transport and tunnel mode. The TOE uses the Encapsulating Security Payload (ESP) protocol to provide authentication and encryption supporting the following algorithms: AES-GCM-128, AES-GCM-192 and AES-GCM-256 together with a Secure Hash Algorithm (SHA)-based HMAC (HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512) for integrity.

The TOE implements IKEv2 and supports the following encryption algorithms for setting up the IKE SA as part of IKEv2 negotiation with a remote peer: AES-GCM-128, AES-GCM-192 and AES-GCM-256. The IKEv2 protocol also supports the following integrity algorithms: HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384 and HMAC-SHA-512.

The IKEv2 protocol also implements DH Group(s) 14 (2048-bit MODP), 15 (3072-bit MODP), 16 (4096-bit MODP), 17 (6144-bit MODP), 18 (8192-bit MODP, 19 (256-bit Random ECP), 20 (384-bit Random ECP) and 21 (521-bit Random ECP). In the IKEv2 IKE_SA and IKEv2 CHILD_SA 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 resulting potential strength of the symmetric key will be 128 or 256 bits of security depending on the algorithms negotiated between the two IPsec peers. As part of this negotiation, the TOE verifies that the negotiated IKEv2 Child SA symmetric algorithm key strength is at most as large as the negotiated IKEv2 IKE SA key strength as configured on the TOE and peer via an explicit check. The TOE checks the IKEv2 IKE SA strength against the IKEv2 Child SA strength and will reject attempts to configure a Child SA with a higher strength.

The TOE supports IKEv2 session establishment and configuration of session lifetimes for both IKEv2 IKE_SAs and IKEv2 Child_SAs. The time values for IKEv2 IKE_SA lifetime SAs can be configured up to 24 hours and for IKEv2 Child_ SAs up to 8 hours. The IKEv2 Child_SA lifetime can also be configured based on number of bytes with configurable values being from between: 1048576 and 4294967295 bytes.

The TOE supports the following authentication mechanisms to authenticate remote IKE peers.

- Pre-shared Key (PSK) - using PSK as the authentication mechanism to authenticate remote IKE peers. The TOE does not generate pre-shared keys, but uses pre-shared keys configured by the authorized administrator. The PSK can be entered as a string with a range of 8 to 128 characters in length (ASCII format) or 8 to 128 bytes in length (Hexadecimal format).
- X.509 certificates (RSA and ECDSA) - using X.509 digital certificates as the authentication mechanism to authenticate remote IKE peers. The local entity certificates used by the local IKE protocol daemon as it's identity, must be configurable by the user.

For peer authentication using RSA and ECDSA certificates, the TOE validates the presented identifier provided supporting the following fields and types: SAN: IPv4 address, SAN: Fully Qualified Domain Name (FQDN) and Distinguished Name (DN). The SAN identifier type is explicitly configured as part of the reference identifier by an authorized administrator.

The TSF generates the secret value 'x' used in the IKEv2 Diffie-Hellman key exchange ('x' in gx mod p) using the NIST approved DRBG specified in FCS_RBG_EXT.1 and having possible lengths of at least

- 224 bits (for DH Group 14),
- 256 bits (for DH Group 15),
- 320 bits (for DH Group 16,
- 384 bits (for DH Group 17),
- 512 bits (for DH Group 18),
- 256 bits (for DH Group 19),
- 384 bits (for DH Group 20), and
- 521 bits (for DH Group 21)

The TOE supports the following PRF hash functions: PRF_HMAC_SHA1, PRF_HMAC_SHA256, PRF_HMAC_SHA384 and PRF_HMAC-SHA512 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 nonce is likewise generated using the TOE's ISO/IEC 18031:2011 AES-256 CTR DRBG.

6.2.10 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, thus ensuring the integrity of the time has been maintained. The TOE does not update the NTP timestamp from broadcast or multicast addresses. The TOE can be configured with up to three NTP. When multiple NTP Servers are configured, the TOE determines which of these Servers to be used as an active source of timing based on the NTP selection and clustering algorithms as specified in RFC 5905 for NTPv4. If the selected NTP Server experiences a fault, the TOE ensures that another of the configured NTP servers is available as a timing source. System time can also be manually configured by an administrator.

6.2.11 FCS_RBG_EXT.1

The TOE supports an AES-CTR 256-bit ISO/IEC 18031:2011 DRBG. The DRBG is seeded by an entropy source that accumulates entropy from a platform-based noise source.

The DRBG is seeded with a minimum of 256 bits of entropy, which is at least equal to the greatest security strength of the keys and hashes that it will generate.

6.2.12 FCS_SSHS_EXT.1

The TOE implements SSHv2 for remote administration via the CLI and NETCONF. The port numbers for CLI and NETCONF are configurable and the default port for CLI is 22 and for NETCONF is 830.

The TOE complies with the following SSHv2 RFCs: 4251, 4252, 4253, 4254, 4256, 5647, 5656, 8308 section 3.1 and 8332. The TOE implements both password-based and public key-based user authentication methods in SSHv2. The TOE supports SSH public-key user authentication using rsa-sha2-256, rsa-sha2-512, ecdsa-sha2-nistp256, ecdsa-sha2-nistp384 and ecdsa-sha2-nistp521. When RSA or ECDSA authentication is used, the TOE checks the presented public key against its authorized keys database and verifies the user's possession of a private key by negotiating a secure channel using the public key associated with that private key.

The TOE checks the packet length of an incoming packet. If a packet greater than 262130 bytes is 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 and aes256-gcm@openssh.com.
- Integrity: hmac-sha2-256, hmac-sha2-512 and implicit (when aes*-gcm@openssh.com is negotiated as the encryption algorithm, the MAC algorithm field is ignored and GCM is implicitly used as the MAC.)
- Host key authentication: rsa-sha2-256, rsa-sha2-512, ecdsa-sha2-nistp256, ecdsa-sha2-nistp384 and ecdsa-sha2-nistp521.
- User public key authentication: rsa-sha2-256, rsa-sha2-512, ecdsa-sha2-nistp256, ecdsa-sha2-nistp384, ecdsa-sha2-nistp521
- Key Exchange: diffie-hellman-group14-sha256, ecdh-sha2-nistp256, ecdh-sha2-nistp384 and ecdh-sha2-nistp521.

The TOE automatically initiates a rekey of the SSH session keys at either 1 hour or 1 gigabyte of traffic. Both thresholds are checked. Rekeying is performed upon reaching whichever threshold is met first.

6.2.13 FCS_TLSS_EXT.1

The TOE supports TLSv1.2 sessions for remote administration via the Web UI. The TOE supports the TLS ciphersuites identified in the FCS_TLSS_EXT.1 requirement in Section 5.1.2.13. The TOE does not support mutual authentication and does not require client authentication.

An authorized administrator can initiate inbound TLSv1.2 connections using the Web UI for remote administration of the TOE. Any session where the client offers the following in the client hello: SSL 2.0, SSL 3.0, TLS 1.0 and TLS 1.1 will be rejected by the TOE's TLS server. If the remote TLS client does not support TLSv1.2 the TLS connections will fail and the administrators will not establish a HTTPS web-based session with the TOE.

The TOE uses RSA or ECDSA X.509 certificates for authentication. The TOE performs key establishment using Diffie-Hellman groups: ffdhe2048, ffdhe3072, ffdhe4096, ffdhe6144, ffdhe8192 or ECDHE curves: secp256r1, secp384r1, secp521r1 depending on the chosen cipher suite. The key agreement parameters of the server key exchange message are specified in the RFC 5246 (section 7.4.3) for TLSv1.2.

The TOE supports TLSv1.2 with AES 128- or 256-bit symmetric ciphers in CBC and GCM modes, in conjunction with SHA, RSA, ECDSA and ECDHE.

The TOE does not support session resumption or session tickets. It is explicitly disabled in the TOE.

6.3 Identification and authentication

The TOE requires all users to be successfully identified and authenticated before allowing any TSF mediated actions to be performed except for acknowledging a pre-login warning banner in accordance with FTA_TAB.1.

6.3.1 FIA_AFL.1

For all methods of remote administration- SSH (CLI, NETCONF) and HTTPS/TLS (Web UI, RESTCONF) - the TOE tracks each unsuccessful authentication attempt for each user account. The number of unsuccessful authentication attempts can be configured between 1 and 255 with the default being 5. The administrator defined lockout time period can be configured from 0 to 1440 minutes. If the counter reaches the administrator specified number of failed login attempts, then the account will be locked out and prevented from gaining access even with a valid password until an administrator specified time period has expired. The account is thereby unlocked upon expiration of the lockout period. The counter is reset to zero upon successful authentication. Failed login attempts are tracked across all remote interfaces with a single counter. Local serial console port access to the CLI is not subject to the lockout and ensures that administrator access is always available.

6.3.2 FIA_PMG_EXT.1

The TOE allows administrators to configure a minimum password length for users that is between 8 and 200 characters. Along with upper and lower case letters and numbers, the TOE allows the following special characters: [*'!', '@', '#', '$', '%', '^', '&', '*', '(', ')' **[and additional special characters: <sp>*** ~ _ - + = ` | { } " [ ] : ; < > , . ? / ' \ ***/***]. 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 a remote RADIUS or TACACS+ database for authenticating users.

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

The TOE requires all administrators to be successfully identified and authenticated before allowing any TSF mediated actions to be performed except for acknowledging the pre-login warning banner. The TOE allows administrators to login to the TOE locally via a directly connected console serial port or to login remotely via SSH (CLI, NETCONF) and via HTTPS/TLS (Web UI, RESTCONF). SSH supports both password and public key authentication, while HTTPS/TLS supports password authentication only.

The process for password authentication is the same for administrative access whether administration is occurring via a directly connected console cable or remotely via SSH or TLS. Once a potential administrative user attempts to access the TOE through either a directly connected console or remotely through SSH or HTTPS/TLS, the TOE prompts the user for a user name and password. Only after the administrative user presents the correct authentication credentials will access to the TOE administrative functionality be granted. The TOE either grants administrative access (if the combination of username and password is correct) or indicates that the login was unsuccessful. No

access is allowed to the administrative functionality of the TOE until an administrator is successfully identified and authenticated.

For SSH remote administration the TOE can also be configured to authenticate using a public key mechanism (RSA or ECDSA). Note that the TOE does not support configuration of both password and public key authentication at the same time. When SSH user public key authentication is configured, the TOE ensures and verifies that the SSH client's presented public key matches one that is stored within the TOE's SSH server's authorized keys file. If a user attempts public key-based authentication and it succeeds, the authentication process is completed and the user is granted access, otherwise, the TOE will indicate that the public key login was unsuccessful and the connection will be rejected.

A successful login to the TOE will present the following:

CLI (local console and SSH) – the command prompt is displayed, e.g. secadmin@GX>

NETCONF- the NETCONF server replies with a hello message and sends its capabilities list and session ID.

Web UI – the WebGUI interface is displayed.

RESTCONF- the RESTCONF server replies with a hello message and sends its capabilities list and session ID.

The TOE also supports authentication using external authentication servers. The TOE communicates with a Remote Authentication Dial-In User Service (RADIUS) server and a Terminal Access Controller Access Control Server (TACACS+) for authentication of configured users. The communication with the RADIUS and TACACS+ servers is protected via IPsec.

When a user enters their password at the local console none of the typed characters are echoed in the password field.

6.3.4 FIA_X509_EXT.1/Rev, 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. The trust chain is a shared set of CAs for both TLS and IPsec operations. The TOE does not support mutual authentication for TLS.

The TOE will perform certificate verification when it receives a certificate from a peer or when a certificate is installed. For IPsec, the TOE 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 the 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. The TOE will not authenticate using only a leaf certificate - full path resolution is always required. If a partial path is provided by the peer, the system will ask for the missing certificates to resolve the path to a known root. If the peer does not provide them then the connection is aborted. If the certificate is invalid, the TOE rejects the connection attempt.

The TOE supports the following Certificate revocation methods: Certificate Revocation List and CRL Distribution Point (CDP) and Online Certificate Status Protocol (OCSP). OCSP and CRL revocation methods can be simultaneously enabled. If both methods are configured, then OCSP is used by default to query the Certificate revocation status. CRL will be used if a definite revocation status is not determined by the OCSP method. The TOE checks the IPsec peer's certificate revocation during the authentication step of a 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 relies on certificate revocation checking by communicating with an external server specified in the certificate's AIA extension for OCSP or CDP extension for CRL. If the TOE cannot establish communications with the external server for OCSP revocation checks, then the TOE will not accept the certificate and will terminate the connection attempt with the peer and there will be no fall back to CRL-based revocation checking. If only CRL revocation checking is enabled and the TOE cannot establish communications with the external server, the TOE will not accept the certificate and will terminate the connection attempt.

The TOE generates Certificate Signing Request (CSR) Messages and includes the following information: public key and Common Name. Upon receiving the CA Certificate response, the TOE will validate the chain of certificates from the Root CA. The TOE supports both RSA and ECDSA CSR generation.

6.4 Security management

6.4.1 FMT_MOF.1/ManualUpdate

The TOE restricts the ability to perform manual software updates to Security Administrators. The Security Administrators can query the software version running on the TOE and can manually initiate updates. The software update consists of downloading the new software image to the TOE, activating the software and rebooting the device.

6.4.2 FMT_MTD.1/CoreData

The TOE requires all users to be successfully identified and authenticated before allowing any TSF mediated actions to be performed. Prior to being granted access, a login warning banner is displayed. Only Security Administrators can manage TSF data. There are no security functions available through any interfaces prior to administrator login.

The term “Security Administrator” used in the ST refers to any user account assigned access privileges (as defined in Section 6.4.5 below) allowing them to perform all TOE security management functions.

The trust store is accessed when Security administrators import/remove and assign certificates to endpoints as described in the Admin Guide. Only TOE Security Administrators have the required access privileges to manage the trust store.

6.4.3 FMT_MTD.1/CryptoKeys

No administrative functionality is available prior to administrative login. Only TOE Security Administrators can control (generate/import/delete) the following keys: RSA and ECDSA key pairs to be used in the TLS, SSH and IPsec protocols. The Admin Guide provides instructions for generating these key pairs and for assigning keys to users for SSH user public key authentication.

6.4.4 FMT_SMF.1

All TOE security functions can be performed via the remote SSH interfaces (CLI, NETCONF) and remote HTTPS/TLS interfaces (Web UI, RESTCONF). All TOE security functions, with the exception of managing cryptographic keys, can also be performed via the local console.

The TOE provides the Security 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 modify the behavior of the transmission of audit data to an external IT entity,
- Ability to configure the list of TOE-provided services available before an entity is identified and authenticated, as specified in FIA_UIA_EXT.1,

- Ability to manage the cryptographic keys,
- Ability to configure the cryptographic functionality,
- Ability to configure the lifetime for IPsec SAs,
- Ability to set the time which is used for time-stamps;
- Ability to configure NTP,
- Ability to configure the reference identifier for the peer;
- Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors,
- Ability to import X509v3 certificates to the TOE's trust store,
- Ability to manage the trusted public keys database

6.4.5 FMT_SMR.2

The TOE maintains the security role of Security Administrator who can manage the TOE via both local and remotely accessible administrator interfaces.

The TOE defines access privileges to restrict user access to resources. Each access privilege allows a specific set of actions to be performed. One or more access privileges can be assigned to each user account.

In the evaluated configuration, the TOE Security Administrator role is a user account that is assigned the following access privilege levels:

- Security Administrator (SA)—allows the user to perform network element security management and administration related tasks.
- Network Administrator (NA)—allows the user to monitor the network element, manage equipment, turn-up network element, provision services, administer various network-related functions such as Auto-discovery and topology.
- Encryption Access (EA)—allows the user to perform data encryption procedures.

Thus, the term “Security Administrator” used in the ST refers to any user assigned the access privileges identified above allowing them to perform all TOE security management functions. Management functions are exclusively restricted to Security Administrators.

6.5 Protection of the TSF

6.5.1 FPT_APW_EXT.1, FPT_SKP_EXT.1

Passwords are stored encrypted in the TOE database using a PBKDF2 (Password-Based Key Derivation Function 2) as detailed in RFC 2898. Passwords are salted and hashed using PBKDF2 with hmac-sha512 and 100000 iterations. Cryptographic keys are stored as described in the table in Section 6.2.3 above. The TOE does not offer any functions that will disclose to any users a stored cryptographic key. The TOE provides no interfaces to read pre-shared, symmetric keys, private keys or passwords.

6.5.2 FPT_STM_EXT.1

The TOE has an internal, real-time system clock that maintains time across reboots and power loss. The internal clock can be manually set or configured to sync with an external NTP source. The clock function provides date and time information which is used for the following security functions:

- To provide a timestamp for audit records

- To support timing elements of cryptographic functions (e.g. certificate validity checks to determine if a certificate is expired and certificate revocation checks)
- Monitoring and measuring local and remote administrative sessions for inactivity
- Unlocking of administrator accounts which have been locked as a result of authentication failure for a specified time period
- Determine when SSH and TLS session keys have expired and to initiate a rekey
- Determine when to renegotiate SAs for IPsec tunnels

6.5.3 FPT_TST_EXT.1

The TOE performs a suite of self-tests to verify the correct operation of the cryptographic module. Software images are cryptographically signed, and an image with an invalid signature will not be loaded by the TOE. If a self-test fails, the TOE goes into "Error" state and disables all access to cryptographic functions and Critical Security Parameters (CSPs). The management interfaces do not respond to any commands until the TOE is operational again. The system will be accessible only through the serial interface. The administrator will need to login to the local console and perform an on demand self-test. If this does not clear the Error state on the system, the administrator must contact Infinera Technical Support.

Self-tests can be performed on demand and periodically during normal operation by powering off and then powering on the TOE device or by executing the 'fips self-test' command. The full suite of self-tests is then executed.

The TOE performs self-tests at power-up to verify the integrity of the firmware images and the correct operation of the FIPS-approved algorithm implementations for AES, SHA, HMAC, ECDSA, RSA, DRBG, and KDFs.

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 via the TOE management interfaces (local console, SSH (CLI, NETCONF) and HTTPS/TLS (Web UI, RESTCONF). For example, the software version can be viewed via System Properties in the Web UI dashboard or by issuing the 'swversion' command at the CLI command-line prompt (local console or SSH). The software update file is downloaded from the Infinera Customer Support Portal (https://support.infinera.com). Signature verification is performed when the image is uploaded to the TOE for both full software package upgrades and delta/patch upgrades. The file's digital signature is verified before it is saved to the TOE's flash. The TOE checks the update image integrity using ECDSA P-521 with SHA2-512. If the integrity verification fails, the TOE does not save the uploaded image to flash and the installation fails. Once the file is verified successfully and saved to the flash, the administrator issues commands or performs actions via the Web UI to install and activate the image. The TOE will then reboot and will come up following the reboot with the newly installed version of the software.

6.6 TOE access

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

An authorized administrator can configure both remote or local session idle timeout based on a time period of inactivity. The session idle timeout is the maximum amount of time an administrator may remain idle and is configured per user from 1 to 1440 minutes. A session (local or remote) that is inactive for the defined time period will be terminated and will require re-identification and authentication to establish a new session.

Authorized administrators can also terminate their own interactive sessions. They can log out of both local and remote CLI administrative sessions by issuing the 'exit' command and from the Web UI via the "logout" icon.

6.6.2 FTA_TAB.1

At each administrative interface, the TOE displays an access banner with an administrator specified advisory notice and consent warning regarding use of the TOE. The banner displays on the local console, SSH CLI and the Web UI (HTTPS/TLS) interfaces prior to allowing any administrative access to the TOE.

6.7 Trusted path/channels

6.7.1 FTP_ITC.1

The TOE protects communications between itself and external authentication (RADIUS, TACACS+), SYSLOG, and NTP servers using IPsec/IKE with pre-shared keys or X509 certificates for authentication and assured identification of the non-TSF endpoint. The TOE can either initiate IKE sessions or respond to IKE INIT requests from IPsec peers.

6.7.2 FTP_TRP.1/Admin

The TOE uses SSHv2 and HTTPS/TLS 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.

All remote administration of the TOE takes place over a secure communication path between the remote administrator and the TOE using either an SSHv2 client or a web browser.

- SSH Client – The remote administrator uses an SSH client to access the CLI and the NETCONF API over a secure, encrypted SSHv2 session.
- Web browser – The remote administrator uses a web browser to access the Web UI and the RESTCONF API over a secure HTTPS/TLS connection.