Gigamon GigaVUE Version 6.0

Security Target

ST Version: 1.0
February 3, 2023

Gigamon, Inc.
3300 Olcott Street
Santa Clara, CA 95054

Prepared By:

Booz | Allen | Hamilton
delivering results that endure

Cyber Assurance Testing Laboratory
1100 West St
Laurel, MD 20707

Table of Contents

List of Figures

List of Tables

1 Security Target Introduction

This chapter presents the Security Target (ST) identification information and an overview. An ST contains the Information Technology (IT) security requirements of an identified Target of Evaluation (TOE) and specifies the functional and assurance security measures offered by the TOE.

1.1 ST Reference

This section provides information needed to identify and control this ST and its Target of Evaluation.

1.1.1 ST Identification

ST Title: Gigamon GigaVUE Version 6.0 Security Target
ST Version: 1.0
ST Publication Date: February 3, 2023
ST Author: Booz Allen Hamilton

1.1.2 Document Organization

Chapter 1 of this document provides identifying information for the ST and TOE as well as a brief description of the TOE and its associated TOE type.

Chapter 2 describes the TOE in terms of its physical boundary, logical boundary, exclusions, and dependent Operational Environment components.

Chapter 3 describes the conformance claims made by this ST.

Chapter 4 describes the threats, assumptions, objectives, and organizational security policies that apply to the TOE.

Chapter 5 defines extended Security Functional Requirements (SFRs) and Security Assurance Requirements (SARs).

Chapter 6 describes the SFRs that are to be implemented by the TSF.

Chapter 7 describes the SARs that will be used to evaluate the TOE.

Chapter 8 provides the TOE Summary Specification, which describes how the SFRs that are defined for the TOE are implemented by the TSF.

1.1.3 Terminology

This section defines the terminology used throughout this ST. The terminology used throughout this ST is defined in Table 1-1 and 1-2. These tables are to be used by the reader as a quick reference guide for terminology definitions.

Term	Definition
Administrator or ‘Admin’	A user who is assigned the ‘Admin’ role on the TOE and has the ability to manage the TSF. Synonymous with Security Administrator.

Table 1-1: Customer Specific Terminology

Term	Definition
Credential	Data that establishes the identity of a user (e.g., a cryptographic key or password).
Operating System (OS)	Software that manages hardware resources and provides services for applications.
Platform	A platform can be an operating system, hardware environment, a software-based execution environment, or some combination of these. These types platforms may also run atop other platforms.
Security Administrator	An authorized administrator role that is authorized to manage the TOE and its data. This TOE defines three separate user roles, but only the most privileged role (Admin) is authorized to manage the TOE's security functionality and is therefore considered to be the Security Administrator for the TOE.
Trusted Channel	An encrypted connection between the TOE and a system in the Operational Environment.
Trusted Path	An encrypted connection between the TOE and the application a Security Administrator uses to manage it (SSH client, terminal client, etc.).
User	In a CC context, any individual who has the ability to access the TOE functions or data.

Table 1-2: CC Specific Terminology

1.1.4 Acronyms

The acronyms used throughout this ST are defined in Table 1-3. This table is to be used by the reader as a quick reference guide for acronym definitions.

Acronym	Definition
AES	Advanced Encryption Standard
API	Application Programming Interface
CA	Certificate Authority
CAVP	Cryptographic Algorithm Verification Program
CC	Common Criteria
CLI	Command-Line Interface
cPP	collaborative Protection Profile
CPU	Central Processing Unit
CRL	Certificate Revocation List
CSP	Content Security Policy
DRBG	Deterministic Random Bit Generator
FTP	File Transfer Protocol
HMAC	Hash-based Message Authentication Code
HTTPS	Hypertext Transfer Protocol Secure
I&A	Identity and Access
IDS	Intrusion Detection System
LDAP	Lightweight Directory Access Protocol
MAC	Message Authentication Code
NIAP	National Information Assurance Partnership
NTP	Network Time Protocol
OCSP	Online Certificate Status Protocol
OS	Operating System
PP	Protection Profile
RAM	Random Access Memory
RBG	Random Bit Generator
RNG	Random Number Generator

RU	Rack Unit
SCP	Secure Copy Protocol
SFR	Security Functional Requirement
SFTP	SSH File Transfer Protocol
SHA	Secure Hash Algorithm
SHS	Secure Hash Standard
SPAN	Switched Port Analyzer
SSH	Secure Shell
ST	Security Target
SVR	Server
TAP	Test Access Port
TFTP	Trivial File Transfer Protocol
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Function
UART	Universal Asynchronous Receiver/Transmitter
UI	User Interface

Table 1-3: Acronym Definition

1.1.5 References

[1] collaborative Protection Profile for Network Devices Version 2.2e 20200327 [NDcPP]
[2] Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and general model, dated April 2017, version 3.1, Revision 5, CCMB-2017-004-001
[3] Common Criteria for Information Technology Security Evaluation – Part 2: Security functional components, dated April 2017, version 3.1, Revision 5, CCMB-2017-004-002
[4] Common Criteria for Information Technology Security Evaluation – Part 3: Security assurance components, dated April 2017, version 3.1, Revision 5, CCMB-2017-004-003
[5] Common Methodology for Information Technology Security Evaluation – Evaluation Methodology, dated April 2017, version 3.1, Revision 5, CCMB-2017-004-004
[6] ISO/IEC 18033-3:2010, Information Technology -- Security techniques -- Encryption algorithms — Part3: Block ciphers
[7] ISO/IEC 10116:2017, Information Technology -- Security techniques -- Modes of operation for an n-bit block cipher
[8] ISO/IEC 14888-3:2016, Information Technology -- Security techniques -- Digital signatures with appendix -- Part 3: Discrete logarithm based mechanisms
[9] ISO/IEC 10118-3:2004, Information Technology -- Security techniques -- Hash-functions -- Part 3: Dedicated hash-functions
[10] ISO/IEC 19772:2009, Information Technology – Security techniques – Authenticated encryption
[11] ISO/IEC 9797-2:2011, Information Technology -- Security techniques -- Message Authentication Codes (MACs) -- Part 2: Mechanisms using a dedicated hash-function
[12] ISO/IEC 18031:2011, Information Technology -- Security techniques -- Random bit generation
[13] FIPS PUB 186-4, Digital Signature Standard (DSS), July 2013
[14] NIST Special Publication 800-56A Revision 3: Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography, April 2018

1.2 TOE Reference

The TOE is the Gigamon GigaVUE Version 6.0 family of products, which includes the following appliance models:

Series	GigaVUE HC Series			GigaVUE TA Series		GigaTAP A Series
Component	**HC3**	**HC2**	**HC1**	**TA25**	**TA200**	**GTAP**
Model Number	GVS-HC301 GVS-HC302	GVS-HC2A1 GVS-HC2A2	GVS-HC101 GVS-HC102	GVS-TAX21-HW GVS-TAX22-HW GVS-TAX21A-HW GVS-TAX22A-HW	GVS-TAC21 GVS-TAC22	GTP-ATX21 GTP-ASF21

Table 1-4: TOE Components

1.3 TOE Overview

The Gigamon GigaVUE Visibility Appliances (also known as GigaVUE) Version 6.0 are network devices that include hardware and software. The GigaVUE's primary functionality is to use the Gigamon Forwarding Policy to receive out-of-band (data plane) copied network data from external sources (TAP or SPAN port) and forward that copied network data to one or many tool ports for packet capture or analyzing tools based on user selected criteria.

GigaVUE is made up of the following three model types:

- GigaVUE HC Series enables greater network traffic visibility into data in motion, minimizes traffic overloads and provides options for deploying both inline and out-of-band security and monitoring tools
- The GigaVUE® TA Series of edge network packet brokers are designed to aggregate multiple network links and feed the combined traffic either to GigaVUE HC Series products, or directly to security and monitoring tools, or both.
- The GigaTAP™ (GTAP) A Series is a line of network TAPs designed with intelligent management capabilities that monitor link states of connected devices and the power state of all sources of power to minimize link downtime on network.

All the GigaVUE models can fulfill the NDcPP2E security requirements individually. Therefore, the evaluated configuration consists of the TOE as a standalone device and is not deployed as a distributed manner. Only the functionality claimed in Section 6 of this Security Target is considered to be within the logical boundary of the TOE.

The following figure depicts the TOE boundary.

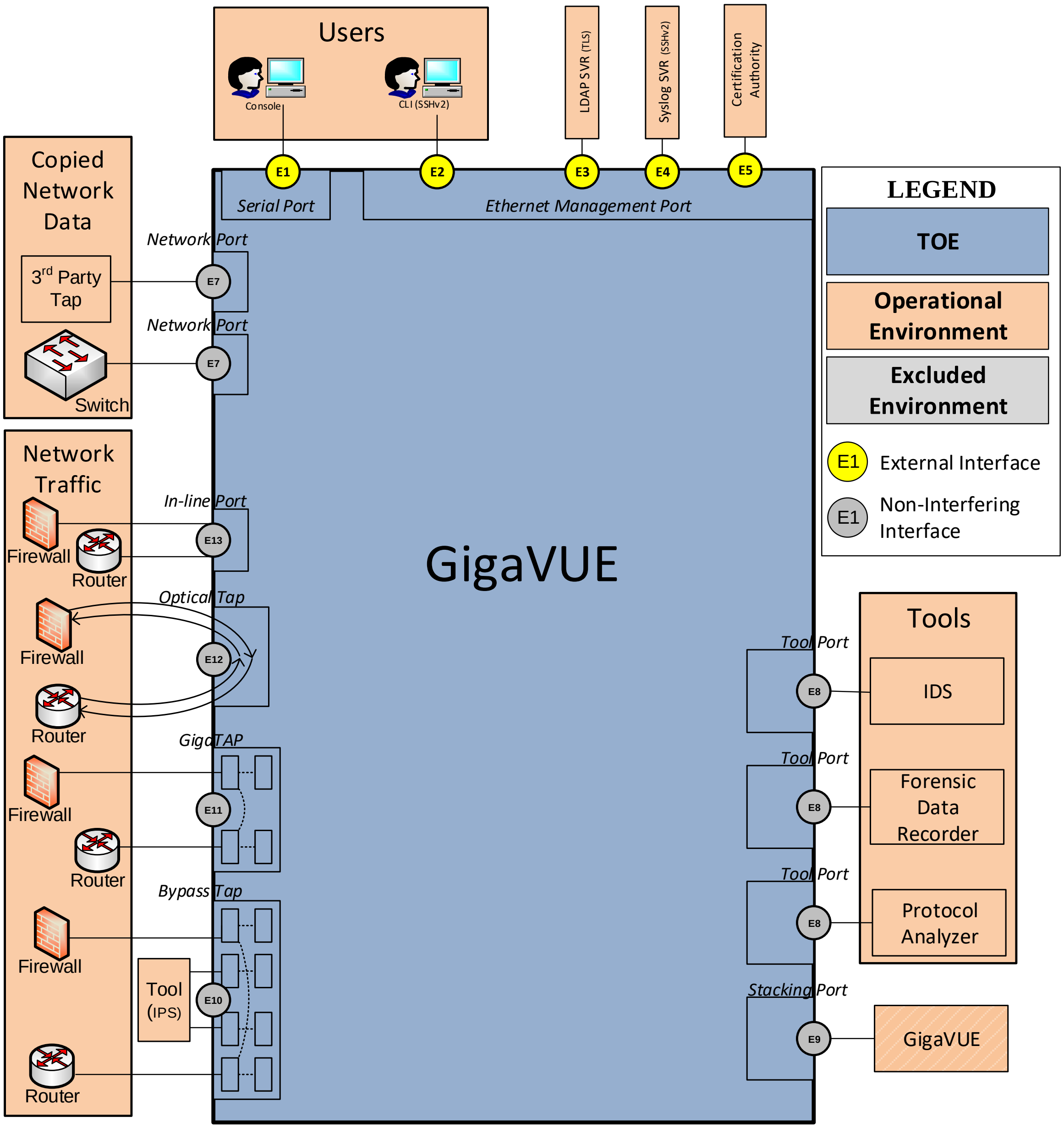

Figure 1: TOE Boundary for GigaVUE

As illustrated in Figure 1, the TOE is a single hardware device that has management ports network (or ingress) ports, and tool (or egress) ports.

The external interfaces that are relevant to the TOE boundary are depicted as an E# in a yellow circle in the figure above. These interfaces are established via a dedicated Management Ethernet Port and collectively is referred to as the management plane. The relevant external interfaces are:

E1: This is the local administrator access to the CLI via a direct connection.

E2: The TOE acts as a SSH server for remote administrator access to the CLI.

E3: The TOE acts as a TLSv1.2 client for accessing an LDAP server interface for authentication services.

E4: The TOE acts as an SSH client for sending audit records to a remote audit server for external audit log storage.

E5: The TOE interfaces with a Certification Authority (CA) for issuance of server certificates and publication of a Certificate Revocation List (CRL) to determine the validity of certificates presented to the TOE.

The following external interfaces are considered non-interfering and are depicted as a E# in a gray circle in the figure above.

E7-E13: These data plane interfaces are used for GigaVUE's primary functionality of forwarding and copying network data to one or many tool ports for packet capture or analyzing tools. The operational functionality performed over the data plane do not map to any NDcPP2E security requirements. Therefore, the non-interfering interfaces and the functionality they provide are not in-scope of the evaluation.

Note: The interface E6 is not identified as it support has been removed from the product.

1.4 TOE Type

The TOE type for this product is Network Device. The product is a hardware appliance whose primary functionality is related to the handling of network traffic.

The NDcPP defines a network device as "a device that is connected to a network and has an infrastructure role within that network…Under this cPP, NDs may be physical or virtualized. A physical Network Device (pND) consists of network device functionality implemented inside a physical chassis with physical network connections. The network device functionality may be implemented in either hardware or software or both. For pNDs, the TOE encompasses the entire device—including both the network device functionality and the physical chassis. There is no distinction between TOE and TOE Platform."

The TOE is a standalone network device composed of hardware and software that is connected to the network and accepts packets of data, filters them and passes them to tools for further analysis. The device's role in the enterprise network is to direct out-of-band network traffic for analysis by various tools, so its infrastructure role is self-evident. There is no distinction between the TOE and the TOE platform.

2 TOE Description

This section provides a description of the TOE in its evaluated configuration. This includes the physical and logical boundaries of the TOE.

2.1 Evaluated Components of the TOE

The following table describes the TOE components and models in the evaluated configuration:

Component	Models Covered	Definition
HC3	GVS-HC301 (AC power) GVS-HC302 (DC power)	3RU Visibility Platform Node (for large enterprises and networks)
HC2	GVS-HC2A1 (AC power) GVS-HC2A2 (DC power)	2RU Visibility Platform Node (for medium-sized networks)
HC1	GVS-HC101 (AC power) GVS-HC102 (DC power)	1RU Visibility Platform Node (for remote and small networks)
TA25	GVS-TAX21-HW (AC power) GVS-TAX22-HW (DC power) GVS-TAX21A-HW (AC power) GVS-TAX22A-HW (DC power	1RU Edge Traffic Aggregation Node (medium to large networks)
TA200	GVS-TAC21 (AC power) GVS-TAC22 (DC power)	2RU Edge Traffic Aggregation Node (medium to large networks)
GTAP	GTP-ASF21 (AC power) GTP-ATX21 (AC power)	1RU Active Network Test Access Point (TAP)

Table 2-1: Evaluated Components of the TOE

2.2 Components and Applications in the Operational Environment

The following table lists components and applications in the TOE's operational environment that must be present for the TOE to be operating in its evaluated configuration:

Component	Definition
Certification Authority (CA)	A server that acts as a trusted issuer of digital certificates and distributes a CRL that identifies revoked certificates.
LDAP Server	A system that is capable of receiving authentication requests using LDAP over TLS and validating these requests against identity and credential data that is defined in an LDAP directory.
Management Workstation	Any general-purpose computer that is used by an administrator to manage the TOE. The TOE can be managed remotely, in which case the management workstation requires an SSH client to access the CLI, or locally, in which case the management workstation must be physically connected to the TOE using the serial port and must use a terminal emulator that is compatible with serial communications.
Audit Server	The audit server connects to the TOE and allows the TOE to send syslog messages to it for remote storage. This is used to send copies of audit data to be stored in a remote location for data redundancy purposes.

Table 2-2: Components of the Operational Environment

2.3 Excluded from the TOE

The following optional products, components, and/or applications can be integrated with the TOE but are not included in the evaluated configuration. They provide no added security related functionality for the evaluated product. They are separated into three categories: not installed, installed but requires a separate license, and installed but not part of the TSF.

2.3.1 Not Installed

There are no optional components that are omitted from the installation process.

2.3.2 Installed but Requires a Separate License

There are no excluded components that are installed and require a separate license.

2.3.3 Installed but Not Part of the TSF

- **Insecure mode of operation** – GigaVUE provides an 'Secure Cryptography Mode' that restricts the cryptographic algorithms and ciphersuites to what is claimed in the Security Target. Operating the product outside of this mode of operation is not within the scope of the TSF
- **TLS mode of syslog handling** – GigaVUE provides an additional method of off-loading the audit data using a TLS connection to the audit server. In the evaluated configuration, only the SSH method of sending audit records is permitted and the remaining methods are not part of the TOE.
- **Port Blades, TAP Modules, Bypass Combo Modules and Port Modules** – Modular components of the GigaVUE used to capture traffic from the network in a variety of methods, for different scenarios, and to support different types of network media. GigaVUE's network traffic capture, filter, and forwarding capabilities were not assessed during this evaluation.
- **Stacking** – Combining multiple GigaVUE devices through a dedicated network port for the purpose of load balancing or to forward traffic received by one device to a tool connected to another device.
- **Telnet** – GigaVUE supports both Telnet and SSH2 for remote administration. In the evaluated configuration Telnet will be disabled.
- **Distributed configuration** – GigaVUE supports forwarding information to other GigaVUE products as part of its operational data plane functions. Each TOE component can fulfill all of the mandated SFRs defined in the NDcPPv2.2E individually and are therefore tested as standalone entities.

Additionally, the TOE includes a number of functions, such as the monitoring and analyzing of network traffic, are outside the scope of the claimed Protection Profile. These functions cannot be mapped to any NDcPPv2.2E SFRs.

2.4 Physical Boundary

2.4.1 Hardware

The GigaVUE HC Series are modular devices to accommodate many variations of physical connectivity including copper, fiber, 1G, 10G, 40G and 100G ports.

The model specific hardware and their configurations are as follows:

Property	HC3	HC2	HC1
Model/Part Number	GVS-HC301 (AC power) GVS-HC302 (DC power)	GVS-HC2A1 (AC power) GVS-HC2A2 (DC power)	GVS-HC101 (AC power) GVS-HC102 (DC power)
Size	3RU	2RU	1RU
Processor	Intel Atom C2758 (Rangeley)	NXP QorIQ P2041E	Intel Atom C2538 (Rangeley)
TAP Modules	None	TAP-HC0-D25AC0 TAP module, SX/SR Internal TAP module 50/125, 12 TAPs TAP-HC0-D25BC0 TAP module, SX/SR Internal TAP module 62.5125, 12 TAPs TAP-HC0-D35CC0 TAP module, LX/LR Internal TAP module, 12 TAPs TAP-HC0-G100C0 TAP and Bypass Module, Copper, 12 TAP or BPS pairs	TAP-HC1-G10040 TAP and Bypass module, 10/100/1000M Copper, 4 TAPs or BPC pairs
Bypass Combo Modules	BPS-HC3-C25F26 Bypass Combo Module, GigaVUE-HC3, 2 100Gb SR4 BPS pairs, 16 10G cages	BPS-HC0-D25A4G Bypass Combo Module 4 SX/SR 50/125 BPS pairs, 16 10G cages BPS-HC0-D25B4G Bypass Combo Module 4 SX/SR 62.5/125 BPS pairs, 16 10G cages BPS-HC0-D35C4G Bypass Combo Module 4 LX/LR BPS pairs, 16 10G cages BPS-HC0-Q25A28 Bypass Combo Module 2 40G SR4 BPS pairs, 8 10G cages	BPS-HC1-D25A24 Bypass Combo Module, 2 SX/SR 50/125 BPS pairs, 4 10G cages
GigaSMART Modules	SMT-HC3-C05 GigaSMART, GigaVUE-HC3, 5x100G QSFP28 cages (includes Slicing, Masking, Source Port Tagging, and	SMT-HC0-R GigaSMART, GigaVUE-HC2 rear module; SMT-HC0-X16 GigaSMART, GigaVUE-HC2 front module, 16 10G	SMT-HC1-S GigaSMART GigaVUE-HC1, Gen3: Processing up to 30G (includes Slicing, Masking, Source Port Tagging, and Tunneling De-

Property	HC3	HC2	HC1
	Tunneling De-encapsulation)	cages (includes Slicing, Masking, Source Port Tagging, and Tunneling De-encapsulation)	encapsulation)
Port Modules	PRT-HC3-C08Q08 Port Module, 8x100G QSFP28 cages, 8x40 QSFP+ cages PRT-HC3-X24 Port Module, GigaVUE-HC3, 24x10G	PRT-HC0-X24 Port Module, 24x10G (QSFP) PRT-HC0-Q06 Port Module, 6x40G (QSFP+) PRT-HC0-C02 Port Module, 2x100G (QSFP28)	None
Fixed Ports	10/100/1000M Mgmt. port Serial Console	10/100/1000M Mgmt. port Serial Console	10/100/1000M Mgmt. port Serial Console 12 1G/10G Ports (QSFP) 4 10/100/1000M Ports
Configurable Ports (provided functionality out of scope as stated in Section 2.3.3)	Provided by Port Modules	Provided by TAP modules, Bypass Combo modules, Port Modules	Provided by TAP modules, Bypass Combo modules

Table 2-3: HC Series Properties

The GigaVUE TA Series are devices to accommodate core traffic aggregation and forwarding needs of medium to large size networks including copper, fiber, 10G, 25G, 40G and 100G networks.

Property	TA25	TA200
Model/Part Number	GVS-TAX21-HW (AC power) • all ports enabled GVS-TAX22-HW (DC power) • all ports enabled GVS-TAX21A-HW (AC power) • 24 10G/25G ports enabled GVS-TAX22A-HW (DC power) • 24 10G/25G ports enabled	GVS-TAC21 (AC power) GVS-TAC22 (DC power)
Size	1RU	2RU
Processor	Intel Atom C3538 (Denverton)	Intel Xeon D1527 (Broadwell)
Fixed Ports	10/100/1000M Mgmt. port Serial Console 8 40G/100G QSFP28 cages + 48 1G/10G/25G SFP28 cages	10/100/1000M Mgmt. port Serial Console 64 100G/40G ports
Configurable Ports	None	None

Table 2-4: TA Series Properties

The GTAP A Series are devices to accommodate core traffic forwarding needs over variations of physical connectivity including 1G, 10G, and 100G copper ports and 1G, 10G fiber ports.

Property	GTAP	
Model/Part Number	GTP-ATX21 (AC power)	GTP-ASF21 (AC power)
Size	1RU	1RU
Processor	Intel Atom C3338 (Denverton)	Intel Atom C3338 (Denverton)
Fixed Ports	10/100/1000M Mgmt. port 4x 10/100/1000BASE-T links	10/100/1000M Mgmt. port 4x 1Gb/10Gb Copper or Fiber links
Configurable Ports	None	None

Table 2-5: GTAP A Series Properties

2.4.2 Software

- Gigamon GigaVUE with software version 6.0.

Note that the GigaVUE software is built on top of the following CentOS versions.

GigaVUE Model	Base OS
HC2	CentOS 5.8
HC1, HC3, TA25, TA200, GTP-ASF21, GTP-ATX21	CentOS 7.6

Table 2-6: Software Versions

2.5 Logical Boundary

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

- Security Audit
- Cryptographic Support
- Identification and Authentication
- Security Management
- Protection of the TSF
- TOE Access
- Trusted Path/Channels

2.5.1 Security Audit

Audit records are generated for various types of management activities and events. The audit records include the date and time stamp of the event, the event type and subject identity. In the evaluated configuration, the TSF is configured to transmit audit data to a remote audit server using SSHv2, but audit data is also stored locally to ensure availability of the data if communications with the audit server are unavailable. Local audit records are stored in “message” files which are rotated to ensure a maximum limit of disk usage is enforced. Only users with the Admin privilege can access or delete the log files.

Users with the Admin privilege are considered trusted users and are therefore not expected to delete or modify the audit records.

2.5.2 Cryptographic Support

The TOE uses sufficient security measures to protect its data in transmission by implementing cryptographic methods and trusted channels. The TOE uses SSH to secure the remote CLI and audit server trusted channels. The TOE also uses TLS to secure the trusted channel for the LDAP server.

Cryptographic keys are generated using the CTR_DRBG provided by this module. The TOE erases all plaintext secret and private keys that reside in both RAM and non-volatile storage with zeroes. In the evaluated configuration, the TOE operates in “Secure Cryptography Mode” which is used to restrict algorithms to meet the PP requirements.

The following table contains the CAVP algorithm certificates:

SFR	Algorithm	CAVP Cert. #
FCS_CKM.1 FCS_COP.1/SigGen	ECDSA	A2202
FCS_CKM.2	KAS-SSC ECC / KAS-ECC CDH	A3221
FCS_COP.1/DataEncryption	AES	A2202
FCS_COP.1/Hash	SHS	A2202
FCS_COP.1/KeyedHash	HMAC	A2202
FCS_RBG_EXT.1	DRBG	A2202

Table 2-7: Cryptographic Algorithm Table

2.5.3 Identification and Authentication

All users must be identified and authenticated to the TOE before being allowed to perform any actions on the TOE. This is true of users accessing the TOE via the local console or the protected path using the remote CLI via SSH. Users authenticate to the TOE using one of the following methods:

- Username/password (defined on the TOE)
- LDAP authentication
- Username/public key (SSH only)

The TSF provides a configurable number of maximum consecutive authentication failures that are permitted by a user. Once this number has been met, the account is locked for a configurable time interval. Passwords that are maintained by the TSF can be composed of upper case, lower case, numbers and special characters. The Security Administrator can define the minimum password length between 8 and 30 characters. Password information is never revealed during the authentication process including during login failures. Before a user authenticates to the device, a configurable warning banner is displayed.

As part of establishing trusted remote communications, the TOE provides X.509 certificate functionality. In addition to verifying the validity of certificates, the TSF can check their revocation status using a certificate revocation list (CRL).

2.5.4 Security Management

The TOE defines two roles: Admin and Monitor. Each of these roles has varying levels of fixed privilege to interact with the TSF. The Admin role is able to perform all security-relevant management functionality (such as user management, password policy configuration, application of software updates, and configuration of cryptographic settings). The Monitor role provides view-only access to ports and configurations. Therefore, the term “Admin”, used throughout this document, is considered to be a Security Administrator of the TSF. Management functions can be performed using the local console or remote CLI. All software updates to the TOE are performed manually.

2.5.5 Protection of the TSF

The TOE stores usernames and passwords in a password file that cannot be viewed by any user on the TOE regardless of the user's role. The passwords are hashed using SHA-512. Public keys are stored in the configuration database which is integrity checked at boot time. Key data is stored in plaintext on the hard drive but cannot be accessed by any user. The TOE has an underlying hardware clock that is used for keeping time. The time can be manually set by the administrator. Power-on self-tests are executed automatically when the cryptographic module is loaded into memory. All binaries (e.g., executables, libraries), are located on a read-only partition and cannot be modified. In addition, the TOE has a configuration database that is integrity checked at boot time.

The version of the TOE (both the currently executing version and the installed/updated version, if different) can be verified from any of the administrative interfaces provided by the TSF. The updated image is verified via a digital signature.

2.5.6 TOE Access

The TOE can terminate inactive local console or remote CLI sessions after a specified time period. The default setting is 15 minutes. Users can also terminate their own interactive sessions. Once a session has been terminated, the TOE requires the user to re-authenticate to establish a new session. The TOE displays an administratively configured banner on the local console or remote CLI prior to allowing any administrative access to the TOE.

2.5.7 Trusted Path/Channels

The TOE connects and sends data to IT entities that reside in the Operational Environment via trusted channels. In the evaluated configuration, the TOE connects with an audit server using SSH to encrypt the audit data that traverses the channel. The TOE also connects with an LDAP server using TLS. When accessing the TOE remotely, administrators interface with the TSF using a trusted path. The remote CLI is protected via SSH.

3 Conformance Claims

3.1 CC Version

This ST is compliant with Common Criteria for Information Technology Security Evaluation, Version 3.1 Revision 5 April 2017.

3.2 CC Part 2 Conformance Claims

This ST and Target of Evaluation (TOE) is Part 2 extended to include all applicable NIAP and International interpretations through February 3, 2023.

3.3 CC Part 3 Conformance Claims

This ST and Target of Evaluation (TOE) are conformant to Part 3 to include all applicable NIAP and International interpretations through February 3, 2023.

3.4 PP Claims

This ST claims exact conformance to the following Protection Profiles:

- collaborative Protection Profile for Network Devices Version 2.2e [NDcPP]

3.5 Package Claims

The TOE claims exact conformance to the NDcPP, which is conformant with CC Part 3.

The TOE claims following Selection-Based SFRs that are defined in the appendices of the claimed PP:

- FCS_SSHC_EXT.1
- FCS_SSHS_EXT.1
- FCS_TLSC_EXT.1
- FIA_X509_EXT.1/Rev
- FIA_X509_EXT.2
- FMT_MTD.1/CryptoKeys

The TOE claims the following Optional SFRs that are defined in the appendices of the claimed PP:

- FAU_STG.1

This does not violate the notion of exact conformance because the NDcPP specifically indicates these as allowable selections and options and provides both the ST author and evaluation laboratory with instructions on how these claims are to be documented and evaluated.

3.6 Package Name Conformant or Package Name Augmented

This ST and TOE are in exact conformance with the NDcPP.

3.7 Conformance Claim Rationale

The NDcPP states the following: "This is a Collaborative Protection Profile (cPP) whose Target of Evaluation (TOE) is a Network Device (ND)… A network device in the context of this cPP is a device connected to the network and has an infrastructure within the network… Examples of network devices that are covered by requirements in this cPP include physical and virtualized routers, firewalls, VPN gateways, IDSs, and switches."

The TOE is a network device composed of hardware and software that is designed to apply forwarding rules to different types of network traffic so that it can be analyzed by a variety of third-party tools. As such, it can be understood as having a role in network infrastructure. Therefore, the conformance claim is appropriate.

3.8 Technical Decisions

TD #	Title	References	Changes			Analysis to this evaluation	
			SFR	AA	Notes	NA	Reason
TD0527	Updates to Certificate Revocation Testing (FIA_X509_EXT.1)	FIA_X509_EXT.1/REV, FIA_X509_EXT.1/ITT		X			AA: Testing Update. No ST updates required.
TD0528	NIT Technical Decision for Missing EAs for FCS_NTP_EXT.1.4	FCS_NTP_EXT.1.4, ND SD v2.2.		X		X	AA: Testing Update. SFR not claimed.
TD0536	NIT Technical Decision for Update Verification Inconsistency	AGD_OPE.1, ND SDv2.2.		X			AA: Guidance Update. No ST updates required.
TD0537	NIT Technical Decision for Incorrect reference to FCS_TLSC_EXT.2.3	FIA_X509_EXT.2.2			X		SFR claimed but Note change has no impact on evaluation documentation.
TD0538	NIT Technical Decision for Outdated link to allowed-with list	Section 2			X		PP claimed but Note change has no impact on evaluation documentation.
TD0546	NIT Technical Decision for DTLS - clarification of Application Note 63	FCS_DTLSC_EXT.1.1			X	X	SFR not claimed
TD0547	NIT Technical Decision for Clarification on developer disclosure of AVA_VAN	ND SDv2.2, AVA_VAN.1		X			Clarification of AVA_VAN No ST updates required.
TD0555	NIT Technical Decision for RFC Reference incorrect in TLSS Test	NDSDv2.2, FCS_TLSS_EXT.1.4, Test 3		X		X	SFR not claimed
TD0556	NIT Technical Decision for RFC 5077 question	NDSDv2.2, FCS_TLSS_EXT.1.4, Test 3		X		X	SFR not claimed
TD0563	NiT Technical Decision for Clarification of audit date information	NDcPPv2.2e, FAU_GEN.1.2			X		Clarified date time stamp requirements No ST updates required. AGD Section 8 shows compliance.

TD0564	NiT Technical Decision for Vulnerability Analysis Search Criteria	NDSDv2.2, AVA_VAN.1			X		Clarified AVA public search requirements.
TD0569	NIT Technical Decision for Session ID Usage Conflict in FCS_DTLSS_EXT.1.7	ND SD v2.2, FCS_DTLSS_EXT.1.7, FCS_TLSS_EXT.1.4		X	X	X	SFR not claimed
TD0570	NiT Technical Decision for Clarification about FIA_AFL.1	FIA_AFL.1			X		Makes FIA_AFL.1 mandatory. FIA_AFL.1 was already claimed. Not marked with footnote as no SFR wording changes were mandated.
TD0571	NiT Technical Decision for Guidance on how to handle FIA_AFL.1	FIA_UAU.1, FIA_PMG_EXT.1			X		Makes FIA_PMG_EXT.1, FIA_AFL.1, and FMT_SMF.1 mandatory. All were previously claimed. Not marked with footnote as no SFR wording changes were mandated.
TD0572	NiT Technical Decision for Restricting FTP_ITC.1 to only IP address identifiers	FTP_ITC.1			X		Clarification; no changes to AA or ST required.
TD0580	NIT Technical Decision for clarification about use of DH14 in NDcPPv2.2e	FCS_CKM.1.1, FCS_CKM.2.1		X	X	X	Not claiming DH14 AA: TSS, Test
TD0581	NIT Technical Decision for Elliptic curve-based key establishment and NIST SP 800-56Arev3	FCS_CKM.2	X				Updated revisions 2 to 3 Footnote 2
TD0591	NIT Technical Decision for Virtual TOEs and hypervisors	A.LIMITED_FUNCTIONALITY, ACRONYMS					A.LIMITED_FUNCTIONALITY wording change. Footnote 1.
TD0592	NIT Technical Decision for Local Storage of Audit Records	FAU_STG					No changes to ST or AA activities. Only changes wording in the PP.
TD0631	NIT Technical Decision for Clarification of public key authentication for SSH Server	ND SDv2.2, FCS_SSHS_EXT.1, FMT_SMF.1	X	X	X		AA: TSS and Tests Footnote 4 for SSHS FCS_SSHS_EXT.1 Footnote 4 for FMT_SMF.1
TD0632	NIT Technical Decision for Consistency with Time Data for vNDs	ND SD2.2, FPT_STM_EXT.1.2	X	X	X	X	AA: TSS, AGD, Test Adds an SFR selection not being claimed. TOE is not vND.
TD0633	NIT Technical Decision for IPsec IKE/SA Lifetimes Tolerance	ND SD2.2, FCS_IPSEC_EXT.1.7, FCS_IPSEC_EXT.1.8		X		X	AA: AGD, Test Not claiming IPsec

TD0634	NIT Technical Decision for Clarification required for testing IPv6	FCS_DTLSC_EXT.1.2, FCS_TLSC_EXT.1.2, ND SD v2.2		X			AA: Test
TD0635	NIT Technical Decision for TLS Server and Key Agreement Parameters	FCS_TLSS_EXT.1.3, NDSD v2.2		X		X	SFR not claimed
TD0636	NIT Technical Decision for Clarification of Public Key User Authentication for SSH	ND SD2.2, FCS_SSHC_EXT.1	X	X	X		AA: TSS and Tests Footnote 3 for SSHS FCS_SSHS_EXT.1
TD0638	NIT Technical Decision for Key Pair Generation for Authentication	NDSDv2.2, FCS_CKM.1			X	X	The TOE does not claim TLS Server functionality and TLS client does not support mutual authentication.
TD0639	NIT Technical Decision for Clarification for NTP MAC Keys	FCS_NTP_EXT.1.2, FAU_GEN.1, FCS_CKM.4, FPT_SKP_EXT.1			X	X	The TOE is not claiming NTP usage
TD0670	NIT Technical Decision for Mutual and Non-Mutual Auth TLSC Testing	ND SD2.2, FCS_TLSC_EXT.2.1		X	X	X	AA: Tests Mutual authentication is not claimed

4 Security Problem Definition

4.1 Threats

This section identifies the threats against the TOE. These threats have been taken from the NDcPP.

Threat	Threat Definition
T.UNAUTHORIZED_ADMINISTRATOR_ ACCESS	Threat agents may attempt to gain Administrator access to the Network Device by nefarious means such as masquerading as an Administrator to the device, masquerading as the device to an Administrator, replaying an administrative session (in its entirety, or selected portions), or performing man-in-the-middle attacks, which would provide access to the administrative session, or sessions between Network Devices. Successfully gaining Administrator access allows malicious actions that compromise the security functionality of the device and the network on which it resides.
T.WEAK_CRYPTOGRAPHY	Threat agents may exploit weak cryptographic algorithms or perform a cryptographic exhaust against the key space. Poorly chosen encryption algorithms, modes, and key sizes will allow attackers to compromise the algorithms, or brute force exhaust the key space and give them unauthorized access allowing them to read, manipulate and/or control the traffic with minimal effort.
T.UNTRUSTED_COMMUNICATION_ CHANNELS	Threat agents may attempt to target Network Devices that do not use standardized secure tunnelling protocols to protect the critical network traffic. Attackers may take advantage of poorly designed protocols or poor key management to successfully perform man-in-the-middle attacks, replay attacks, etc. Successful attacks will result in loss of confidentiality and integrity of the critical network traffic, and potentially could lead to a compromise of the Network Device itself.
T.WEAK_AUTHENTICATION_ ENDPOINTS	Threat agents may take advantage of secure protocols that use weak methods to authenticate the endpoints, e.g. a shared password that is guessable or transported as plaintext. The consequences are the same as a poorly designed protocol, the attacker could masquerade as the Administrator or another device, and the attacker could insert themselves into the network stream and perform a man-in-the-middle attack. The result is the critical network traffic is exposed and there could be a loss of confidentiality and integrity, and potentially the Network Device itself could be compromised.
T.UPDATE_COMPROMISE	Threat agents may attempt to provide a compromised update of the software or firmware which undermines the security functionality of the device. Non-validated updates or updates validated using non-secure or weak cryptography leave the update firmware vulnerable to surreptitious alteration.
T.UNDETECTED_ACTIVITY	Threat agents may attempt to access, change, and/or modify the security functionality of the Network Device without Administrator awareness. This could result in the attacker finding an avenue (e.g., misconfiguration, flaw in the product) to compromise the device and the Administrator would have no knowledge that the device has been compromised.
T.SECURITY_FUNCTIONALITY_ COMPROMISE	Threat agents may compromise credentials and device data enabling continued access to the Network Device and its critical data. The compromise of credentials includes replacing existing

Threat	Threat Definition
	credentials with an attacker's credentials, modifying existing credentials, or obtaining the Administrator or device credentials for use by the attacker.
T.PASSWORD_CRACKING	Threat agents may be able to take advantage of weak administrative passwords to gain privileged access to the device. Having privileged access to the device provides the attacker unfettered access to the network traffic and may allow them to take advantage of any trust relationships with other Network Devices.
T.SECURITY_FUNCTIONALITY_ FAILURE	An external, unauthorized entity could make use of failed or compromised security functionality and might therefore subsequently use or abuse security functions without prior authentication to access, change or modify device data, critical network traffic or security functionality of the device.

Table 4-1: TOE Threats

4.2 Organizational Security Policies

This section identifies the organizational security policies which are expected to be implemented by an organization that deploys the TOE. These policies have been taken from the NDcPP.

Policy	Policy Definition
P.ACCESS_BANNER	The TOE shall display an initial banner describing restrictions of use, legal agreements, or any other appropriate information to which users consent by accessing the TOE.

Table 4-2: Organizational Security Policies

4.3 Assumptions

The specific conditions listed in this section are assumed to exist in the TOE's Operational Environment. These assumptions have been taken from the NDcPP.

Assumption	Assumption Definition
A.PHYSICAL_PROTECTION	The Network Device is assumed to be physically protected in its operational environment and not subject to physical attacks that compromise the security or interfere with the device's physical interconnections and correct operation. This protection is assumed to be sufficient to protect the device and the data it contains. As a result, the cPP does not include any requirements on physical tamper protection or other physical attack mitigations. The cPP does not expect the product to defend against physical access to the device that allows unauthorized entities to extract data, bypass other controls, or otherwise manipulate the device. For vNDs, this assumption applies to the physical platform on which the VM runs.
A.LIMITED_FUNCTIONALITY[1]	The device is assumed to provide networking functionality as its core function and not provide functionality/services that could be deemed as general purpose computing. For example, the device should not provide a computing platform for general purpose applications (unrelated to networking functionality).

[1] TD0591

Assumption	Assumption Definition
	If a virtual TOE evaluated as a pND, following Case 2 vNDs as specified in Section 1.2, the VS is considered part of the TOE with only one vND instance for each physical hardware platform. The exception being where components of a distributed TOE run inside more than one virtual machine (VM) on a single VS. In Case 2 vND, no non-TOE guest VMs are allowed on the platform.
A.NO_THRU_TRAFFIC_ PROTECTION	A standard/generic Network Device does not provide any assurance regarding the protection of traffic that traverses it. The intent is for the Network Device to protect data that originates on or is destined to the device itself, to include administrative data and audit data. Traffic that is traversing the Network Device, destined for another network entity, is not covered by the ND cPP. It is assumed that this protection will be covered by cPPs and PP-Modules for particular types of Network Devices (e.g., firewall).
A.TRUSTED_ADMINISTRATOR	The Security Administrator(s) for the Network Device are assumed to be trusted and to act in the best interest of security for the organization. This includes appropriately trained, following policy, and adhering to guidance documentation. Administrators are trusted to ensure passwords/credentials have sufficient strength and entropy and to lack malicious intent when administering the device. The Network Device is not expected to be capable of defending against a malicious Administrator that actively works to bypass or compromise the security of the device. For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are expected to fully validate (e.g. offline verification) any CA certificate (root CA certificate or intermediate CA certificate) loaded into the TOE's trust store (aka 'root store', ' trusted CA Key Store', or similar) as a trust anchor prior to use (e.g. offline verification).
A.REGULAR_UPDATES	The Network Device firmware and software is assumed to be updated by an Administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
A.ADMIN_CREDENTIALS_ SECURE	The Administrator's credentials (private key) used to access the Network Device are protected by the platform on which they reside.
A.RESIDUAL_INFORMATION	The Administrator must ensure that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment.

Table 4-3: TOE Assumptions

4.4 Security Objectives

This section identifies the security objectives of the TOE and its supporting environment. The security objectives identify the responsibilities of the TOE and its environment in meeting the security needs.

4.4.1 TOE Security Objectives

The NDcPP does not define any security objectives for the TOE.

4.4.2 Security Objectives for the Operational Environment

The TOE's operational environment must satisfy the following objectives:

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

Table 4-4: Operational Environment Objectives

4.5 Security Problem Definition Rationale

The assumptions, threats, OSPs, and objectives that are defined in this ST represent the assumptions, threats, OSPs, and objectives that are specified in the Protection Profile to which the TOE claims conformance.

5 Extended Components Definition

5.1 Extended Security Functional Requirements

The extended Security Functional Requirements that are claimed in this ST are taken directly from the PP to which the ST and TOE claim conformance. These extended components are formally defined in the PP in which their usage is required.

5.2 Extended Security Assurance Requirements

The extended Security Assurance Requirements that are claimed in this ST are taken directly from the PP to which the ST and TOE claim conformance. These extended components are formally defined in the PP in which their usage is required.

6 Security Functional Requirements

6.1 Conventions

The CC permits four functional component operations—assignment, refinement, selection, and iteration—to be performed on functional requirements. This ST will highlight the operations in the following manner:

- **Assignment:** allows the specification of an identified parameter. Indicated with *italicized* text.
- **Refinement:** allows the addition of details. Indicated with **bold** text. Note that conversion of British spelling to American spelling is not marked as a refinement (e.g., 'authorisation' changed to 'authorization').
- **Selection:** allows the specification of one or more elements from a list. Indicated with <u>underlined</u> text.
- **Iteration:** allows a component to be used more than once with varying operations. Indicated with a sequential number in parentheses following the element number of the iterated SFR and/or separated by a "/" with a notation that references the function for which the iteration is used, e.g., "/TrustedUpdate" for an SFR that relates to update functionality

When multiple operations are combined, such as an assignment that is provided as an option within a selection or refinement, a combination of the text formatting is used.

If SFR text is reproduced verbatim from text that was formatted in a claimed PP (such as if the PP's instantiation of the SFR has a refinement or a completed assignment), the formatting is not preserved. This is so that the reader can identify the operations that are performed by the ST author as opposed to the PP author.

6.2 Security Functional Requirements Summary

The following table lists the SFRs claimed by the TOE:

Class Name	Component Identification	Component Name
Security Audit (FAU)	FAU_GEN.1	Audit Data Generation
	FAU_GEN.2	User Identity Association
	FAU_STG.1	Protected Audit Trail Storage
	FAU_STG_EXT.1	Protected Audit Event Storage
Cryptographic Support (FCS)	FCS_CKM.1	Cryptographic Key Generation
	FCS_CKM.2	Cryptographic Key Establishment
	FCS_CKM.4	Cryptographic Key Destruction
	FCS_COP.1/DataEncryption	Cryptographic Operation (AES Data Encryption/Decryption)
	FCS_COP.1/SigGen	Cryptographic Operation (Signature Generation and Verification)
	FCS_COP.1/Hash	Cryptographic Operation (Hash Algorithm)
	FCS_COP.1/KeyedHash	Cryptographic Operation (Keyed Hash Algorithm)
	FCS_RBG_EXT.1	Cryptographic Operation (Random Bit Generation)
	FCS_SSHC_EXT.1	SSH Client Protocol

Class Name	Component Identification	Component Name
	FCS_SSHS_EXT.1	SSH Server Protocol
	FCS_TLSC_EXT.1	TLS Client Protocol Without Mutual Authentication
Identification and Authentication (FIA)	FIA_AFL.1	Authentication Failure Management
	FIA_PMG_EXT.1	Password Management
	FIA_UAU_EXT.2	Password-based Authentication Mechanism
	FIA_UAU.7	Protected Authentication Feedback
	FIA_UIA_EXT.1	User Identification and Authentication
	FIA_X509_EXT.1/Rev	X509 Certificate Validation
	FIA_X509_EXT.2	X509 Certificate Authentication
Security Management (FMT)	FMT_MOF.1/ManualUpdate	Management of Security Functions Behavior
	FMT_MTD.1/CoreData	Management of TSF Data
	FMT_MTD.1/CryptoKeys	Management of TSF Data
	FMT_SMF.1	Specification of Management Functions
	FMT_SMR.2	Restrictions on Security Roles
Protection of the TSF (FPT)	FPT_APW_EXT.1	Protection of Administrator Passwords
	FPT_SKP_EXT.1	Protection of TSF Data (For Reading of All Pre-shared, Symmetric and Private Keys)
	FPT_STM_EXT.1	Reliable Time Stamps
	FPT_TST_EXT.1	TSF Testing
	FPT_TUD_EXT.1	Trusted Update
TOE Access (FTA)	FTA_SSL_EXT.1	TSF-initiated Session Locking
	FTA_SSL.3	TSF-Initiated Termination
	FTA_SSL.4	User-initiated Termination
	FTA_TAB.1	TOE Access Banners
Trusted Path/Channels (FTP)	FTP_ITC.1	Inter-TSF Trusted Channel
	FTP_TRP.1/Admin	Trusted Path

Table 6-1: Security Functional Requirements for the TOE

6.3 Security Functional Requirements

6.3.1 Class FAU: Security Audit

6.3.1.1 *FAU_GEN.1 Audit Data Generation*

FAU_GEN.1.1

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

a) Start-up and shut-down of the audit functions;
b) All auditable events for the 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 **6-2**.

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 **6-2**.

Requirement	Auditable Event(s)	Additional Audit Record Contents
FAU_GEN.1	None.	None.
FAU_GEN.2	None.	None.
FAU_STG.1	None.	None.
FAU_STG_EXT.1	None.	None.
FCS_CKM.1	None.	None.
FCS_CKM.2	None.	None.
FCS_CKM.4	None.	None.
FCS_COP.1/DataEncryption	None.	None.
FCS_COP.1/SigGen	None.	None.
FCS_COP.1/Hash	None.	None.
FCS_COP.1/KeyedHash	None.	None.
FCS_RBG_EXT.1	None.	None.
FCS_SSHC_EXT.1	Failure to establish an SSH session.	Reason for failure.
FCS_SSHS_EXT.1	Failure to establish an SSH session.	Reason for failure.
FCS_TLSC_EXT.1	Failure to establish a TLS session.	Reason for failure.
FIA_AFL.1	Unsuccessful login attempts limit is met or exceeded.	Origin of the attempt (e.g., IP Address).
FIA_PMG_EXT.1	None.	None.
FIA_UAU_EXT.2	All use of identification and authentication mechanism.	Origin of the attempt (e.g., IP Address).
FIA_UAU.7	None.	None.
FIA_UIA_EXT.1	All use of identification and authentication mechanism.	Origin of the attempt (e.g., IP Address).
FIA_X509_EXT.1/Rev	Unsuccessful attempt to validate a certificate. Any addition, replacement or removal of trust anchors in the TOE's trust store.	Reason for failure of certificate validation. Identification of certificates added, replaced or removed as trust anchor in the TOE's trust store.
FIA_X509_EXT.2	None.	None.
FMT_MOF.1/ManualUpdate	Any attempt to initiate a manual update**.**	None.
FMT_MTD.1/CoreData	None.	None.
FMT_MTD.1/CryptoKeys	None.	None.
FMT_SMF.1	All management activities of TSF data.	None.
FMT_SMR.2	None.	None.
FPT_APW_EXT.1	None.	None.

FPT_SKP_EXT.1	None.	None.
FPT_STM_EXT.1	Discontinuous changes to time – either Administrator actuated or changed via an automated process. (Note that no continuous changes to time need to be logged. See also application note on FPT_STM_EXT.1)	For discontinuous changes to time: The old and new values for the time. Origin of the attempt to change time for success and failure (e.g., IP address).
FPT_TST_EXT.1	None.	None.
FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure).	None.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	None.
FTA_SSL.4	The termination of an interactive session.	None.
FTA_SSL_EXT.1	The termination of a local session by the session locking mechanism.	None.
FTA_TAB.1	None.	None.
FTP_ITC.1	Initiation of the trusted channel. Termination of the trusted channel. Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempt.
FTP_TRP.1/Admin	Initiation of the trusted path. Termination of the trusted path. Failure of the trusted path functions.	None.

Table 6-2: Auditable Events

6.3.1.2 ***FAU_GEN.2*** ***User Identity Association***

FAU_GEN.2.1

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

6.3.1.3 ***FAU_STG.1*** ***Protected Audit Trail Storage***

FAU_STG.1.1

The TSF shall protect the stored audit records in the audit trail from unauthorized deletion.

FAU_STG.1.2

The TSF shall be able to prevent unauthorized modifications to the stored audit records in the audit trail.

6.3.1.4 ***FAU_STG_EXT.1*** ***Protected Audit Event Storage***

FAU_STG_EXT.1.1

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

FAU_STG_EXT.1.2

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

- The TOE shall consist of a single standalone component that stores audit data locally].

FAU_STG_EXT.1.3

The TSF shall [overwrite previous audit records according to the following rule: [*rotate compressed messages files*]] when the local storage space for audit data is full.

6.3.2 Class FCS: Cryptographic Support

6.3.2.1 *FCS_CKM.1* *Cryptographic Key Generation*

FCS_CKM.1.1

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

- 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].

6.3.2.2 *FCS_CKM.2* *Cryptographic Key Establishment*

FCS_CKM.2.1[2]

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"].

6.3.2.3 *FCS_CKM.4* *Cryptographic Key Destruction*

FCS_CKM.4.1

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

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

that meets the following: No Standard.

[2] TD0581

6.3.2.4 *FCS_COP.1/DataEncryption Cryptographic Operation (AES Data Encryption/Decryption)*

FCS_COP.1.1/DataEncryption

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

6.3.2.5 *FCS_COP.1/SigGen Cryptographic Operation (Signature Generation and Verification)*

FCS_COP.1.1/SigGen

The TSF shall perform cryptographic signature services (generation and verification) in accordance with a specified cryptographic algorithm [

- Elliptic Curve Digital Signature Algorithm and cryptographic key sizes [*256 bits*]]

that meet the following: [

- 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].

6.3.2.6 *FCS_COP.1/Hash Cryptographic Operation (Hash Algorithm)*

FCS_COP.1.1/Hash

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

6.3.2.7 *FCS_COP.1/KeyedHash Cryptographic Operation (Keyed Hash Algorithm)*

FCS_COP.1.1/KeyedHash

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

6.3.2.8 *FCS_RBG_EXT.1 Cryptographic Operation (Random Bit Generation)*

FCS_RBG_EXT.1.1

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

FCS_RBG_EXT.1.2

The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [[*2*] software-based noise source] with a minimum of [256 bits] of entropy at least equal to the greatest security strength, according to ISO/IEC 18031:2011 Table C.1 "Security Strength Table for Hash Functions", of the keys and hashes that it will generate.

6.3.2.9 *FCS_SSHC_EXT.1* *SSH Client Protocol*

FCS_SSHC_EXT.1.1

The TSF shall implement the SSH protocol in accordance with: RFCs 4251, 4252, 4253, 4254, [5656, 6668].

FCS_SSHC_EXT.1.2[3]

The TSF shall ensure that the SSH protocol implementation supports the following user authentication methods as described in RFC 4252: public key-based, [no other method].

FCS_SSHC_EXT.1.3

The TSF shall ensure that, as described in RFC 4253, packets greater than [*32,768*] bytes in an SSH transport connection are dropped.

FCS_SSHC_EXT.1.4

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

FCS_SSHC_EXT.1.5

The TSF shall ensure that the SSH public-key based authentication implementation uses [ecdsa-sha2-nistp256, ecdsa-sha2-nistp384, ecdsa-sha2-nistp521] as its public key algorithm(s) and rejects all other public key algorithms.

FCS_SSHC_EXT.1.6

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

FCS_SSHC_EXT.1.7

The TSF shall ensure that [ecdh-sha2-nistp256] and [ecdh-sha2-nistp384, ecdh-sha2-nistp521] are the only allowed key exchange methods used for the SSH protocol.

FCS_SSHC_EXT.1.8

[3] TD0636

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.

FCS_SSHC_EXT.1.9

The TSF shall ensure that the SSH client authenticates the identity of the SSH server using a local database associating each host name with its corresponding public key and [no other methods] as described in RFC 4251 section 4.1.

6.3.2.10 *FCS_SSHS_EXT.1* *SSH Server Protocol*

FCS_SSHS_EXT.1.1

The TSF shall implement the SSH protocol in accordance with: RFCs 4251, 4252, 4253, 4254, [5656, 6668].

FCS_SSHS_EXT.1.2[4]

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

FCS_SSHS_EXT.1.3

The TSF shall ensure that, as described in RFC 4253, packets greater than [*32,768*] bytes in an SSH transport connection are dropped.

FCS_SSHS_EXT.1.4

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

FCS_SSHS_EXT.1.5

The TSF shall ensure that the SSH public-key based authentication implementation uses [ecdsa-sha2-nistp384] as its public key algorithm(s) and rejects all other public key algorithms.

FCS_SSHS_EXT.1.6

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

FCS_SSHS_EXT.1.7

The TSF shall ensure that [ecdh-sha2-nistp256] and [ecdh-sha2-nistp384, ecdh-sha2-nistp521] are the only allowed key exchange methods used for the SSH protocol.

FCS_SSHS_EXT.1.8

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

[4] TD0631

6.3.2.11 *FCS_TLSC_EXT.1* *TLS Client Protocol Without Mutual Authentication*

FCS_TLSC_EXT.1.1

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

- TLS_ECDHE_ECDSA_WITH_AES_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]

and no other ciphersuites.

FCS_TLSC_EXT.1.2

The TSF shall verify that the presented identifier matches [the reference identifier per RFC 6125 section 6 and no other attribute types].

FCS_TLSC_EXT.1.3

When establishing a trusted channel, by default the TSF shall not establish a trusted channel if the server certificate is invalid. The TSF shall also [

- Not implement any administrator override mechanism].

FCS_TLSC_EXT.1.4

The TSF shall [present the Supported Elliptic Curves/Supported Groups Extension with the following curves/groups: [secp256r1, secp384r1, secp521r1] and no other curves/groups] in the Client Hello.

6.3.3 Class FIA: Identification and Authentication

6.3.3.1 *FIA_AFL.1* *Authentication Failure Management*

FIA_AFL.1.1

The TSF shall detect when an Administrator configurable positive integer within [*1 to 4,294,967,296*] unsuccessful authentication attempts occur related to Administrators attempting to authenticate remotely using a password.

FIA_AFL.1.2

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

6.3.3.2 *FIA_PMG_EXT.1* *Password Management*

FIA_PMG_EXT.1.1

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

a) Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters: [“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, “)”];
b) Minimum password length shall be configurable to between [*8*] and [*30*] characters.

6.3.3.3 ***FIA_UAU_EXT.2*** ***Password-Based Authentication Mechanism***

FIA_UAU_EXT.2.1

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

6.3.3.4 ***FIA_UAU.7*** ***Protected Authentication Feedback***

FIA_UAU.7.1

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

6.3.3.5 ***FIA_UIA_EXT.1*** ***User Identification and Authentication***

FIA_UIA_EXT.1.1

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

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

FIA_UIA_EXT.1.2

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

6.3.3.6 ***FIA_X509_EXT.1/Rev*** ***X.509 Certificate Validation***

FIA_X509_EXT.1.1/Rev

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

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

- Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.
- Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.
- Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.
- OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.

FIA_X509_EXT.1.2/Rev

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

6.3.3.7 *FIA_X509_EXT.2* *X.509 Certificate Authentication*

FIA_X509_EXT.2.1

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

FIA_X509_EXT.2.2

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

6.3.4 Class FMT: Security Management

6.3.4.1 *FMT_MOF.1/ManualUpdate* *Management of Security Functions Behavior*

FMT_MOF.1.1/ManualUpdate

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

6.3.4.2 *FMT_MTD.1/CoreData* *Management of TSF Data*

FMT_MTD.1.1/CoreData

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

6.3.4.3 *FMT_MTD.1/CryptoKeys* *Management of TSF Data*

FMT_MTD.1.1/CryptoKeys

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

6.3.4.4 *FMT_SMF.1* *Specification of Management Functions*

FMT_SMF.1.1[5]

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 manage the cryptographic keys;
  - Ability to configure the cryptographic functionality;
  - Ability to set the time which is used for time-stamps;
  - Ability to manage the trusted public keys database].

6.3.4.5 *FMT_SMR.2* *Restrictions on Security Roles*

FMT_SMR.2.1

The TSF shall maintain the roles:

- Security Administrator.

FMT_SMR.2.2

The TSF shall be able to associate users with roles.

FMT_SMR.2.3

The TSF shall ensure that the conditions

- The Security Administrator role shall be able to administer the TOE locally;
- The Security Administrator role shall be able to administer the TOE remotely

are satisfied.

6.3.5 Class FPT: Protection of the TSF

6.3.5.1 *FPT_APW_EXT.1* *Protection of Administrator Passwords*

FPT_APW_EXT.1.1

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

FPT_APW_EXT.1.2

The TSF shall prevent the reading of plaintext administrative passwords.

---

[5] TD0631

6.3.5.2 *FPT_SKP_EXT.1* *Protection of TSF Data (For Reading of All Pre-Shared, Symmetric and Private Keys)*

FPT_SKP_EXT.1.1

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

6.3.5.3 *FPT_STM_EXT.1* *Reliable Time Stamps*

FPT_STM_EXT.1.1

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

FPT_STM_EXT.1.2

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

6.3.5.4 *FPT_TST_EXT.1* *TSF Testing*

FPT_TST_EXT.1.1

The TSF shall run a suite of the following self-tests [during initial start-up (on power on)] to demonstrate the correct operation of the TSF: [*software integrity, cryptographic module integrity, hardware integrity, continuous RNG test*].

6.3.5.5 *FPT_TUD_EXT.1* *Trusted Update*

FPT_TUD_EXT.1.1

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

FPT_TUD_EXT.1.2

The TSF shall provide Security Administrators the ability to manually initiate updates to TOE firmware/software and [no other update mechanism].

FPT_TUD_EXT.1.3

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

6.3.6 Class FTA: TOE Access

6.3.6.1 *FTA_SSL_EXT.1* *TSF-Initiated Session Locking*

FTA_SSL_EXT.1.1

The TSF shall, for local interactive sessions, [

- terminate the session]

after a Security Administrator-specified time period of inactivity.

6.3.6.2	*FTA_SSL.3*	*TSF-Initiated Termination*

FTA_SSL.3.1

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

6.3.6.3	*FTA_SSL.4*	*User-Initiated Termination*

FTA_SSL.4.1

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

6.3.6.4	*FTA_TAB.1*	*TOE Access Banner*

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.

6.3.7 Class FTP: Trusted Path/Channels

6.3.7.1	*FTP_ITC.1*	*Inter-TSF Trusted Channel*

FTP_ITC.1.1

The TSF shall be capable of using [SSH, TLS] to provide a trusted communication channel between itself and authorized IT entities supporting the following capabilities: audit server, [authentication 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.

FTP_ITC.1.2

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

FTP_ITC.1.3

The TSF shall initiate communication via the trusted channel for [*authentication requests, transferring audit records*].

6.3.7.2	*FTP_TRP.1/Admin*	*Trusted Path*

FTP_TRP.1.1/Admin

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

FTP_TRP.1.2/Admin

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

FTP_TRP.1.3/Admin

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

6.4 Statement of Security Functional Requirements Consistency

The Security Functional Requirements included in the ST represent all required SFRs specified in the claimed PP, a subset of the optional requirements, and all applicable selection-based requirements that have been included as specified for the claimed PP.

7 Security Assurance Requirements

This section identifies the Security Assurance Requirements (SARs) that are claimed for the TOE. The SARs which are claimed are in exact conformance with the NDcPP.

Security Target (ASE)	ST introduction (ASE_INT.1)
	Conformance claims (ASE_CCL.1)
	Security Problem Definition (ASE_SPD.1)
	Security objectives for the operational environment (ASE_OBJ.1)
	Extended components definition (ASE_ECD.1)
	Stated security requirements (ASE_REQ.1)
	TOE summary specification (ASE_TSS.1)
Development (ADV)	Basic functional specification (ADV_FSP.1)
Guidance Documents (AGD)	Operational user guidance (AGD_OPE.1)
	Preparative procedures (AGD_PRE.1)
Life Cycle Support (ALC)	Labelling of the TOE (ALC_CMC.1)
	TOE CM coverage (ALC_CMS.1)
Tests (ATE)	Independent testing – conformance (ATE_IND.1)
Vulnerability Assessment (AVA)	Vulnerability survey (AVA_VAN.1)

7.1 Class ASE: Security Target evaluation

7.1.1 ST introduction (ASE_INT.1)

7.1.1.1 *Developer action elements:*

ASE_INT.1.1D

The developer shall provide an ST introduction.

7.1.1.2 *Content and presentation elements:*

ASE_INT.1.1C

The ST introduction shall contain an ST reference, a TOE reference, a TOE overview and a TOE description.

ASE_INT.1.2C

The ST reference shall uniquely identify the ST.

ASE_INT.1.3C

The TOE reference shall uniquely identify the TOE.

ASE_INT.1.4C

The TOE overview shall summarise the usage and major security features of the TOE.

ASE_INT.1.5C

The TOE overview shall identify the TOE type.

ASE_INT.1.6C

The TOE overview shall identify any non-TOE hardware/software/firmware required by the TOE.

ASE_INT.1.7C

The TOE description shall describe the physical scope of the TOE.

ASE_INT.1.8C

The TOE description shall describe the logical scope of the TOE.

7.1.1.3 *Evaluator action elements:*

ASE_INT.1.1E

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

ASE_INT.1.2E

The evaluator shall confirm that the TOE reference, the TOE overview, and the TOE description are consistent with each other.

7.1.2 Conformance claims (ASE_CCL.1)

7.1.2.1 *Developer action elements:*

ASE_CCL.1.1D

The developer shall provide a conformance claim.

ASE_CCL.1.2D

The developer shall provide a conformance claim rationale

7.1.2.2 *Content and presentation elements:*

ASE_CCL.1.1C

The conformance claim shall contain a CC conformance claim that identifies the version of the CC to which the ST and the TOE claim conformance.

ASE_CCL.1.2C

The CC conformance claim shall describe the conformance of the ST to CC Part 2 as either CC Part 2 conformant or CC Part 2 extended.

ASE_CCL.1.3C

The CC conformance claim shall describe the conformance of the ST to CC Part 3 as either CC Part 3 conformant or CC Part 3 extended.

ASE_CCL.1.4C

The CC conformance claim shall be consistent with the extended components definition.

ASE_CCL.1.5C

The conformance claim shall identify all PPs and security requirement packages to which the ST claims conformance.

ASE_CCL.1.6C

The conformance claim shall describe any conformance of the ST to a package as either package-conformant or package-augmented.

ASE_CCL.1.7C

The conformance claim rationale shall demonstrate that the TOE type is consistent with the TOE type in the PPs for which conformance is being claimed.

ASE_CCL.1.8C

The conformance claim rationale shall demonstrate that the statement of the security problem definition is consistent with the statement of the security problem definition in the PPs for which conformance is being claimed.

ASE_CCL.1.9C

The conformance claim rationale shall demonstrate that the statement of security objectives is consistent with the statement of security objectives in the PPs for which conformance is being claimed.

ASE_CCL.1.10C

The conformance claim rationale shall demonstrate that the statement of security requirements is consistent with the statement of security requirements in the PPs for which conformance is being claimed.

7.1.2.3 *Evaluator action elements:*

ASE_CCL.1.1E

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

7.1.3 Security problem definition (ASE_SPD)

7.1.3.1 *Developer action elements:*

ASE_SPD.1.1D

The developer shall provide a security problem definition.

7.1.3.2 *Content and presentation elements:*

ASE_SPD.1.1C

The security problem definition shall describe the threats.

ASE_SPD.1.2C

All threats shall be described in terms of a threat agent, an asset, and an adverse action.

ASE_SPD.1.3C

The security problem definition shall describe the OSPs.

ASE_SPD.1.4C

The security problem definition shall describe the assumptions about the operational environment of the TOE.

7.1.3.3 *Evaluator action elements:*

ASE_SPD.1.1E

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

7.1.4 Security objectives for the operational environment (ASE_OBJ.1)

7.1.4.1 *Developer action elements:*

ASE_OBJ.1.1D

The developer shall provide a statement of security objectives.

7.1.4.2 *Content and presentation elements:*

ASE_OBJ.1.1C

The statement of security objectives shall describe the security objectives for the operational environment.

7.1.4.3 *Evaluator action elements:*

ASE_OBJ.1.1E

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

7.1.5 Extended components definition (ASE_ECD.1)

7.1.5.1 *Developer action elements:*

ASE_ECD.1.1D

The developer shall provide a statement of security requirements.

ASE_ECD.1.2D

The developer shall provide an extended components definition.

7.1.5.2 *Content and presentation elements:*

ASE_ECD.1.1C

The statement of security requirements shall identify all extended security requirements.

ASE_ECD.1.2C

The extended components definition shall define an extended component for each extended security requirement.

ASE_ECD.1.3C

The extended components definition shall describe how each extended component is related to the existing CC components, families, and classes.

ASE_ECD.1.4C

The extended components definition shall use the existing CC components, families, classes, and methodology as a model for presentation.

ASE_ECD.1.5C

The extended components shall consist of measurable and objective elements such that conformance or nonconformance to these elements can be demonstrated.

7.1.5.3 *Evaluator action elements:*

ASE_ECD.1.1E

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

ASE_ECD.1.2E

The evaluator shall confirm that no extended component can be clearly expressed using existing components.

7.1.6 Stated security requirements (ASE_REQ.1)

7.1.6.1 *Developer action elements:*

ASE_REQ.1.1D

The developer shall provide a statement of security requirements.

ASE_REQ.1.2D

The developer shall provide a security requirements rationale.

7.1.6.2 ***Content and presentation elements:***

ASE_REQ.1.1C

The statement of security requirements shall describe the SFRs and the SARs.

ASE_REQ.1.2C

All subjects, objects, operations, security attributes, external entities and other terms that are used in the SFRs and the SARs shall be defined.

ASE_REQ.1.3C

The statement of security requirements shall identify all operations on the security requirements.

ASE_REQ.1.4C

All operations shall be performed correctly.

ASE_REQ.1.5C

Each dependency of the security requirements shall either be satisfied, or the security requirements rationale shall justify the dependency not being satisfied.

ASE_REQ.1.6C

The statement of security requirements shall be internally consistent.

7.1.6.3 ***Evaluator action elements:***

ASE_REQ.1.1E

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

7.1.7 TOE summary specification (ASE_TSS.1)

7.1.7.1 ***Developer action elements:***

ASE_TSS.1.1D

The developer shall provide a TOE summary specification.

7.1.7.2 ***Content and presentation elements:***

ASE_TSS.1.1C

The TOE summary specification shall describe how the TOE meets each SFR.

7.1.7.3 ***Evaluator action elements:***

ASE_TSS.1.1E

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

ASE_TSS.1.2E

The evaluator shall confirm that the TOE summary specification is consistent with the TOE overview and the TOE description.

7.2 Class ADV: Development

7.2.1 Basic Functional Specification (ADV_FSP.1)

7.2.1.1 *Developer action elements:*

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.

7.2.1.2 *Content and presentation elements:*

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.

7.2.1.3 *Evaluator action elements:*

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.

7.3 Class AGD: Guidance Documentation

7.3.1 Operational User Guidance (AGD_OPE.1)

7.3.1.1 *Developer action elements:*

AGD_OPE.1.1D

The developer shall provide operational user guidance.

7.3.1.2 *Content and presentation elements:*

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.

7.3.1.3 *Evaluator action elements:*

AGD_OPE.1.1E

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

7.3.2 Preparative Procedures (AGD_PRE.1)

7.3.2.1 *Developer action elements:*

AGD_PRE.1.1D

The developer shall provide the TOE including its preparative procedures.

7.3.2.2 *Content and presentation elements:*

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.

7.3.2.3 *Evaluator action elements:*

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.

7.4 Class ALC: Life Cycle Supports

7.4.1 Labeling of the TOE (ALC_CMC.1)

7.4.1.1 *Developer action elements:*

ALC_CMC.1.1D

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

7.4.1.2 *Content and presentation elements:*

ALC_CMC.1.1C

The TOE shall be labeled with its unique reference.

7.4.1.3 *Evaluator action elements:*

ALC_CMC.1.1E

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

7.4.2 TOE CM Coverage (ALC_CMS.1)

7.4.2.1 *Developer action elements:*

ALC_CMS.1.1D

The developer shall provide a configuration list for the TOE.

7.4.2.2 *Content and presentation elements:*

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.

7.4.2.3 *Evaluator action elements:*

ALC_CMS.1.1E

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

7.5 Class ATE: Tests

7.5.1 Independent Testing - Conformance (ATE_IND.1)

7.5.1.1 *Developer action elements:*

ATE_IND.1.1D

The developer shall provide the TOE for testing.

7.5.1.2 *Content and presentation elements:*

ATE_IND.1.1C

The TOE shall be suitable for testing.

7.5.1.3 *Evaluator action elements:*

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.

7.6 Class AVA: Vulnerability Assessment

7.6.1 Vulnerability Survey (AVA_VAN.1)

7.6.1.1 *Developer action elements:*

AVA_VAN.1.1D

The developer shall provide the TOE for testing.

7.6.1.2 *Content and presentation elements:*

AVA_VAN.1.1C

The TOE shall be suitable for testing.

7.6.1.3 *Evaluator action elements:*

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.

8 TOE Summary Specification

The following sections identify the security functions of the TOE and describe how the TSF meets each claimed SFR. They include Security Audit, Cryptographic Support, Identification and Authentication, Security Management, Protection of the TSF, TOE Access, and Trusted Path/Channels.

8.1 Security Audit

8.1.1 FAU_GEN.1:

The TOE contains mechanisms to generate audit data based upon successful and unsuccessful management actions by all authorized users of the TOE. Each audit record contains identifying information of the subject performing the action. The audit records are generated and stored in the form of syslog records which are sent securely to the audit server protected by SSHv2. The TOE maintains log levels which determine the set of events that are logged. The log level is set using the following command: `logging level`. Setting the log-level to "info" captures all the necessary logs defined in the NDcPP.

The TOE allows viewing of the audit records through the local console with the following command: `show log.` Users of any role can view audit log files, however, only Admin users can delete audit log files. If an Admin deletes a log file, an audit record of that action is also recorded.

The auditable events include the start-up and shut-down of the audit functions; administrative actions including login, logout, TSF configuration changes, managing cryptographic keys and resetting passwords; and all events defined in Table 6-2. Audit records for cryptographic functions, such as generating/import of, changing, or deleting cryptographic keys, will contain the value that represents the key to identify the key. Examples of audit records for these events can be found in the Supplemental Administrative Guidance Document.

The audit records that the TOE creates include the following information: date and time of the event (year, month, day, hour, minute, sec), event type, subject identity, and success or failure of the event. Additionally, specific events require additional information as defined by the 'Additional Audit Record Contents' column of Table 6-2.

8.1.2 FAU_GEN.2:

The TOE records the identity of the user (e.g., username, system name, IP address) associated with each audited event in the audit record.

8.1.3 FAU_STG.1:

The TOE allows viewing of the audit records through the CLI with the following command: `show log`. Users of any role can view audit log files, however, only Admin users can delete audit log files. No modification of log files is permitted, regardless of role. If an Admin deletes a log file, an audit record of that action is also recorded. The TOE allocates up to 8 MB for the audit log but rolls this data over to a backup file and compresses it once full. A maximum of 8 backup files exist, and the oldest one is deleted as part of a rollover process when a new backup is created.

8.1.4 FAU_STG_EXT.1:

In the evaluated configuration, the TOE will send audit records to a remote audit server via an encrypted SSH channel over the Ethernet Management Port. When the audit server is configured, the audit records are stored locally and immediately pushed to the audit server. If the audit server connectivity is unavailable, audit records will only be stored locally. Upon re-establishment of communications with the audit server, new audit records will resume being transmitted to it but the audit records that were generated during the time the audit server connection was down remain stored locally and are not sent to the audit server. This is a standalone TOE that is responsible for storing and sending its own generated audit records.

New audit records are stored locally on the TOE under the /var/log directory in the file named “messages". The "message" file is archived when it reaches a specific size (8 MB) by compressing it and saving the file as "messages.1.gz”. Meanwhile, a new "messages" file is created for new audit records and the other compressed messages files are rotated so that the 8 most recent compressed messages files are saved. The 8 compressed files are named "messages.1.gz”, “messages2.gz”, and so on. Therefore, as part of the file rotation “messages8.gz” will be deleted, "messages.7.gz" will be saved as "messages.8.gz", "messages.6.gz" will be saved as "messages.7.gz", and so on until the "messages" file is compressed into "messages.1.gz”. This mechanism guarantees a maximum limit of disk usage used by the log files. Only a user with the Admin role can delete the log files. Users with the Admin role are considered trusted users and are not expected to delete the audit records.

8.2 Cryptographic Support

8.2.1 FCS_CKM.1:

The TOE generates ECC keys using NIST curve P-256, P-384, and P-521, in accordance with FIPS PUB 186-4. The ECC keys are generated in support of device authentication for TLS and SSH.

The TOE’s key generation function is validated under CAVP ECDSA certificate: #A2202.

8.2.2 FCS_CKM.2:

The TOE implements NIST SP 800-56A Revision 3 conformant key establishment mechanisms for Elliptic Curve Diffie-Hellman (ECDH) key establishment schemes. Specifically, the TOE complies with the NIST SP 800-56A Revision 3 key agreement scheme (KAS) primitives that are defined in section 5.6 of the SP. This is used for the establishment of TLS sessions for which the TOE can act as a TLS client and SSH sessions for which the TOE can act as a SSH client and server.

The TOE’s implementation of NIST SP 800-56A is validated under CAVP KAS-SSC ECC/KAS-ECC CDH certificate #A3221.

The following table provides an overview of the usage for each scheme:

Scheme	SFR	Service
ECDH	FCS_TLSC_EXT.1.1	LDAP authentication
ECDH	FCS_SSHC_EXT.1.7	Audit server connection
ECDH	FCS_SSHS_EXT.1.7	CLI administration

Table 8-1: Cryptographic Key Establishment Scheme Usage

8.2.3 FCS_CKM.4:

The TOE implements secure key destruction as follows:

- Keys stored in volatile memory: These keys are immediately zeroized using the function memset() upon deallocation. These keys are destroyed when sessions are closed.
- Keys stored in non-volatile memory: The TOE zeroizes all plaintext secret and private cryptographic keys in persistent storage by overwriting the file with zeroes and performing a read verify. Upon successful completion of the zeroization, the file is deallocated using the file system API unlink(). These keys are destroyed during import/re-installation or upgrade/regeneration.

The TOE is not subject to any situations that would prevent or delay key destruction and strictly conforms to the key destruction requirements. This combined approach protects the keys in volatile and non-volatile memory from being compromised. The following table identifies the keys and CSPs that are applicable to the TOE as well as their usage, storage location, and method of destruction:

Key Material	Origin	Storage Location	Clearing of Key Material
SSH keys	SSH server/client application	Non-volatile storage/file system	Upon regeneration of keys.
Authentication keys	X.509 certificates	Non-volatile storage/file system	Upon import/creation of configuration DB on installation or upgrade.
TLS session keys	LDAP client	RAM	Destroyed on close of session

Table 8-2: Cryptographic Materials, Storage, and Destruction Methods

8.2.4 FCS_COP.1/DataEncryption:

The TOE performs encryption and decryption using the AES algorithm in CBC and GCM mode with key sizes of 128 and 256 bits. This algorithm implementation is validated under CAVP AES certificate # #A2202. The AES algorithm meets ISO 18033-3, the CBC mode implementation meets ISO 10116, and the GCM mode implementation meets ISO 19772.

8.2.5 FCS_COP.1/SigGen:

The TOE performs signature generation and validation using Elliptic Curve Digital Signature Algorithm (ECDSA). The TOE supports ECDSA with 256-bit key size and implements the NIST P-256, P-384, and P-521 curves. The ECDSA implementation meets ISO/IEC 14888-3 Section 6.4 and FIPS PUB 186-4. This implementation is validated under CAVP ECSDA certificate #A2202.

8.2.6 FCS_COP.1/Hash:

The TOE provides cryptographic hashing services using SHA-256, SHA-384, and SHA-512 with message digest sizes of 256, 384, and 512 bits respectively, as specified in ISO/IEC 10118-3:2004. The TSF uses hashing services the following functions:

- SHA-256, and SHA-512 for SSH data integrity
- SHA-256, and SHA-384 for TLS
- SHA-256, SHA-384, and SHA-512 for TLS NIST curves

- SHA-256, SHA-384, and SHA-512 for HMAC
- SHA-256 for software integrity in support of FPT_TST_EXT.1
- SHA-256 for digital signature
- SHA-512 for password hashing

The SHA algorithm meets ISO/IEC 10118-3:2004 and is validated under CAVP SHS certificate #A2202.

8.2.7 FCS_COP.1/KeyedHash:

The TOE provides keyed-hashing message authentication services using HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512. The HMAC implementation supports key sizes that are equal to block sizes. HMAC is implemented as specified in ISO/IEC 9797-2:2011 Section 7 “MAC Algorithm 2”, and the following MAC sizes are supported:

- HMAC-SHA-256: [key size: 256 bits, digest size: 256 bits, block size: 256 bits, MAC output length: 256 bits] for SSH and TLS
- HMAC-SHA-384: [key size: 384 bits, digest size: 384 bits, block size: 384 bits, MAC output length: 384 bits] for TLS
- HMAC-SHA-512: [key size: 512 bits, digest size: 512 bits, block size: 512 bits, MAC output length: 512 bits] for SSH

The algorithm is validated under CAVP HMAC certificate #A2202. When the TOE uses AES in GCM mode for SSH, the keyed-hashing message authentication is implicit through the selection of AES-GCM for the data integrity MAC algorithm.

8.2.8 FCS_RBG_EXT.1:

The TOE implements a NIST-approved deterministic random bit generator (DRBG). The DRBG used by the TOE is the CTR_DRBG with AES as specified by ISO/IEC 18031:2011. The TOE models uniformly provide two software-based entropy sources as described in the proprietary entropy specification. The DRBG is seeded with a minimum of 256 bits of entropy so that it is sufficient to ensure full entropy for 256-bit keys, which are the largest keys generated by the TSF. The TOE’s DRBG implementation meets ISO/IEC 18031:2011 and is validated under CAVP certificate #A2202.

8.2.9 FCS_SSHC_EXT.1/ FCS_SSHS_EXT.1:

SSHv2 is used to secure the remote CLI management connection (SSH server) and the audit server connection (SSH client) between the TOE and a remote audit server. The traffic for both of these connections is sent via the Ethernet Management Port and by default, the SSHv2 port used is port 22.

The TOE implements the SSHv2 protocol that complies with the following RFCs: 4251, 4252, 4253, 4254, 5656, and 6668. The TOE, when acting as a server, supports password-based and public key-based user authentication methods as described in RFC 4252; both authentication methods are supported on the remote CLI. When TOE acts as TLS server only public key-based user authentication is used when communicating with the remote audit server.

The TOE’s SSH client implementation only supports the use and generation of ecdsa-sha2-nistp384 algorithm for the public key user authentication. While the SSH server implementation allows the use of ecdh-sha2-nistp256, ecdh-sha2-nistp384, and ecdh-sha2-nistp521 for public key user authentication. As an SSH server, the TOE verifies the SSH client's presented public key matches one that is stored within the TOE's authorized-keys file.

Regardless of whether the TOE is acting as an SSH client or SSH server, all SSHv2 connections will be dropped upon detection of any packet greater than 32,768 bytes being transported, as described in RFC 4253.

Data encryption is provided by the AES-CBC-128, AES-CBC-256, aes128-gcm@openssh.com, and aes256-gcm@openssh.com encryption algorithms.

The TOE's SSH client implementation supports ecdsa-sha2-nistp256, ecdsa-sha2-nistp384, and ecdsa-sha2-nistp521 for the host-key public key algorithms and will reject all others. The TOE's SSH server implementation only supports ecdsa-sha2-nistp384 for the host-key public key algorithms and will reject all others.

The algorithms HMAC-SHA2-256, and HMAC-SHA2-512 are used for the data integrity MAC algorithms, except when aes128-gcm@openssh.com or aes256-gcm@openssh.com are the encryption algorithms selected. When the TOE uses AES in GCM mode for SSH, the keyed-hashing message authentication is implicit through the selection of AES-GCM for the data integrity MAC algorithm.

The key exchange methods used in SSHv2 are ecdh-sha2-nistp256, ecdh-sha2-nistp384, and ecdh-sha2-nistp521.

Session keys are created when the TOE establishes an SSHv2 connection. The TOE will monitor the time period during which the SSHv2 session keys are active and how much data has been transmitted using them. The TOE has been hard coded to initiate a rekey when the session keys have been used for hour (3600 seconds) or when 256 MB of data has been transmitted when TOE acts as a client and one hour or 1 GB when the TOE acts as a server. Rekeying is performed upon reaching the threshold that is hit first.

Per RFC 4251 section 4.1, the TOE's SSH client implementation will authenticate the identity of the audit server (i.e., SSH server) by using its local database (i.e., ~/.ssh/known_hosts) which associates each host name with its corresponding public key.

The SSH server's host-key public/private key pairs are generated using the following command:

```
ssh server host-key generate
```

The SSH client's user authentication public/private key pairs are generated using the following command:

```
ssh client user admin identity ecdsa generate
```

The TOE has a "Secure Cryptography Mode" to limit the SSH connection parameters to those defined in the evaluated configuration.

8.2.10 FCS_TLSC_EXT.1:

The TOE uses the TLS 1.2 protocol to secure the LDAP server connection (TLS Client) used for authentication requests. The TOE will reject all connection attempts from TLS versions other than 1.2. When the TOE is operating in "Secure Cryptography Mode", TLS uses the following ciphersuites:

- 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

When the TOE uses TLS Client functionality, the presented identifier for the server certificate has to match the reference identifier in order to establish the connection, per RFC 6125 Section 6. The hostname reference identifier is the only supported value for the LDAP. Wildcards cannot be defined as part of the reference identifier on the TOE, but the TOE will accept certificates with wildcards in the left-most label (e.g. *.example.com). The TOE supports the SAN extensions for certificate validation. The only Supported Elliptic Curves Extension included in the Client Hello are the NIST curves secp256r1, secp384r1, and secp521r1. This is not configurable. Certificate pinning is not supported. When certificate validation fails, the connection is not established.

8.3 Identification and Authentication

8.3.1 **FIA_AFL.1:**

The TSF provides a configurable counter for consecutive failed authentication attempts that will lock an Admin or user account when the failure counter threshold is reached. When an account is locked, an Admin or user cannot login to the remote CLI. A valid login that happens prior to the failure counter reaching its threshold will reset the counter to zero.

The remote CLI counter can be set to any 32-bit integer value (a value of 0 will disable lockout). While the Admin or user account is locked, no authentication is possible. The authentication failure settings can be configured such that the default ‘admin’ user account overrides this functionality (exempt) so that it is not possible to cause a denial of service. The lockout duration is a configurable number of seconds, with a default setting of 360.

8.3.2 **FIA_PMG_EXT.1:**

Passwords maintained by the TSF can be composed using any combination of upper case and lower case letters, numbers, and special characters including the following: “!”,”@”,”#”,”$”,”%”,”^”,”&”,”*”,”(“,”)”. The password policy is configurable by the Admin and supports the minimum password length of 8 characters to 30 characters.

8.3.3 **FIA_UAU_EXT.2:**

Users can authenticate to the TOE locally or remotely. Local users log in to the local console using a username and password via the Serial Port. Remote users can log in to the TOE via the remote CLI using username and password or SSH public key via the Ethernet Management Port. User authentication information that is sent remotely via the remote CLI is protected using SSHv2. When authenticating using username and password, these credentials are verified using either the TOE's local mechanism and credential repository or by an LDAP server that provides external authentication decisions. For external authentication decisions, the TOE sends an authentication request to the LDAP server which will verify the credentials against the LDAP server's directory and will provide an authentication decision back to the TOE, which is subsequently enforced. The connection between the TOE and the LDAP server is protected by TLS v1.2. When public key authentication is used, the TOE authenticates users by verifying the message the TOE receives from the SSH client using the message's associated public key stored on the TOE.

8.3.4 FIA_UAU.7:

While authenticating locally to the TOE, the user's password does not appear in the password field. Instead, asterisks will appear thus masking the password to prevent the password from being shared. In the case that a user enters invalid credentials (valid/invalid username or valid/invalid password), the TOE does not reveal any information about the invalid component of the credential.

8.3.5 FIA_UIA_EXT.1:

In the evaluated configuration, the warning banner is displayed prior to the user authenticating to the TOE via the local console or the remote CLI. The display of the warning banner is the only service that can be run prior to authentication and thus, the TOE does not allow a user to perform any other actions prior to authentication, regardless of the interface used.

Access is only granted once the user provides a valid username/password that is verified using the username/password credential authentication mechanism stated in FIA_UAU_EXT.2.

8.3.6 FIA_X509_EXT.1 and FIA_X509_EXT.2

The TOE performs certificate validity checking for outbound TLS connections to the LDAP Server. In addition to the validity checking that is performed by the TOE, the TSF will validate certificate revocation status using a certificate revocation list (CRL) that the TSF is configured to download automatically from a Certification Authority in the Operational Environment. In the event that the revocation status cannot be verified, the certificate will not be accepted.

The TSF determines the validity of certificates by ensuring that the certificate and the certificate path are valid in accordance with RFC 5280. In addition, the certificate path is terminated in a trusted CA certificate, the basicConstraints extension is present, and the CA flag is set to TRUE for all CA certificates. The TSF also ensures that the extendedKeyUsage field includes the correct purpose for its intended use. This includes Server Authentication for TLS server certificates; the TSF does not handle TLS client certificates, certificates associated with OCSP responses, or code signing certificates.

8.4 Security Management

8.4.1 FMT_MOF.1/ManualUpdate:

The software update must be loaded onto the TOE and applied manually. The Admin role is the only administrative role that can perform this action. The TSF restricts the access to this function by enforcing the GigaVUE's role-based access control system.

8.4.2 FMT_MTD.1/CoreData:

GigaVUE provides two user roles: Admin and Monitor. The TSF uses role-based access control to assign each user account to one or more roles, each of which has a fixed set of privileges to interact with the product. Of these roles, only the Admin role is authorized to perform the management functions associated with the TSF. The Monitor role provides view-only access to ports and configurations. The Admin role is therefore functionally identical to the 'Security Administrator' as defined by the NDcPP.

The only security-relevant TOE functionality that is available to a user prior to authentication is the display of the warning banner.

8.4.3 FMT_MTD.1/CryptoKeys:

The Admin role is the only role that is permitted to manipulate cryptographic data on the TOE. Cryptographic management functions are performed using the CLI commands. Within the TSF, this behavior is limited to the generating/import and deleting of X.509 certificates and SSH keys.

8.4.4 FMT_SMF.1:

A user with the Admin role is capable of performing management functions on the TOE locally and remotely. Admins can perform management functions via the local console or remotely via the remote CLI. The following table lists the TSF management functions and identifies the interface(s) that can be used to perform them:

Management Function	Local Console	Remote CLI
View Audit Data	X	X
Delete Audit Log	X	X
Configure TLS Connection Parameters	X	X
Configure SSH Connection Parameters	X	X
Configure Failed Lockout Threshold	X	X
Configure Lockout Duration	X	X
Create Users	X	X
Modify User Passwords	X	X
Modify Password Policy	X	X
Configure Supported Authentication Mechanism	X	X
Initiate Manual Update	X	X
Configure System Time	X	X
Configure Idle Session Timeout	X	X
Configure Banner Text	X	X
Manage the Cryptographic Keys	X	X
Manage the Trusted Public Keys Database	X	X

Table 8-3: Management Functions by Interface

An Admin can modify the text displayed in the TOE's login banners, set the values for session inactivity before termination, and initiate manual updates to the TOE's software after verifying the digital signature of the update.

The Admin is also able to configure the TOE’s system time (which is used for time-stamps) and its cryptographic functionality. Configuring the cryptographic functionality is accomplished by entering a command into the CLI which places the TOE into "Secure Cryptography Mode" of operation. When this command is entered, the TOE limits the cryptographic algorithms to meet the requirements defined within the Security Target.

8.4.5 FMT_SMR.2:

The security management functions available to authorized users of the TOE are mediated by a role-based access control system. The role-based access control system is enforced via the local console and remotely via the remote CLI. The TOE has two roles: Admin and Monitor. Each role has different authorizations in terms of the functions that they can perform. All SFR relevant management activity is performed by the Admin, a role which corresponds to the NDcPP’s definition of Security Administrator.

Only users with the Admin role are permitted to create and assign roles to users. The Monitor role provides view-only access to ports and configurations.

Each user has the following security attributes associated with them:

- Username
- Password
- SSH public key (optional)
- One or more roles

The username and password are for authenticating to the TOE. These credentials are verified using the authentication mechanism that has been configured for the TOE. Once the username has been validated, the username is used to query the one or more roles which have been associated with that username within the TOE's local store. The TOE then uses the roles assigned to the authenticated user to determine if an action is authorized per GigaVUE's role-based access control system. When LDAP authentication is used, that user information is mapped to the internally-stored attributes so that the authentication event is associated with the correct user.

8.5 Protection of the TSF

8.5.1 **FPT_APW_EXT.1:**

The TOE stores usernames and passwords in a password file. All passwords stored on the TOE are stored in hashed form using the cryptographic hashing services using SHA-512. The password file cannot be viewed by any user on the TOE regardless of the user's role.

8.5.2 **FPT_SKP_EXT.1:**

Public keys are stored in the configuration database which is integrity-checked at boot time. Secret/private key data is stored in plaintext on the hard drive but cannot be accessed via the local console or remote CLI by any user.

8.5.3 **FPT_STM_EXT.1:**

The TOE has an underlying hardware clock that is used for keeping time. A user with the Admin role can set the clock's time manually. The TSF uses time data for the following purposes:

- Audit record timestamps
- Inactivity timeout for administrative sessions
- Expiration checking for certificates
- FIA_AFL.1 timer for lockout duration

8.5.4 **FPT_TST_EXT.1:**

All binaries (e.g., executables, libraries), are located on a read-only partition and cannot be modified. In addition, the TOE has a configuration database that is integrity checked at boot time using SHA-256. The udiag is run under u-boot (microcode boot loader) which runs power-on self-tests of all the major components (e.g., memory, CPU, UART, Ethernet controllers) on the motherboard, including the components that connect to the i2c buses. This includes all transceivers used by the data plane. The pci_diag component is a Linux component that runs when the kernel is loading that is responsible for testing and checking the components connected to the PCIe interfaces. It is also responsible for Line card type detection.

Once booted, the TSF will execute a continuous RNG test in order to ensure that the entropy source has not degraded.

If any of the integrity tests fail, diagnostic information is displayed on the boot console indicating what the problem is. For example:

```
POST Failed
Power-up self test failed
cryptographic algorithm test failed
SW integrity test failed
```

When integrity test fails the TOE is put into safe mode (Gigamon specific state). In safe mode, the device will operate in a limited manner which requires user intervention to bring the appliance back into a normal state after fixing the issues. The console display clearly indicates that the appliance is in safe mode along with the diagnostic information.

These tests are sufficient to validate the correct operation of the TSF because they verify that the cryptographic module is operating correctly, the configuration database does an integrity check, and that the underlying hardware does not have any anomalies that would cause the software to be executed in an unpredictable or inconsistent manner.

8.5.5 **FPT_TUD_EXT.1:**

TOE Admin users can query the current executing version and the most recently installed version of the TOE's firmware/software on the local console or remote CLI interface. A user can enter the "`show version`" command to show both versions of the TOE's firmware/software.

In order to update the TOE, the Admin will access a Gigamon-hosted site and enter a username and password to download the image to their local machine. After downloading the image, the Admin will fetch the image through the remote CLI.

The image that is downloaded is compressed and stored in a .tar file and signed with a digital signature (SHA-256). All GigaVUEs are pre-loaded with a key for the signature verification performed as part of the update mechanism. Before the actual installation occurs, the signature is verified against the stored key. The image will not be installed if the update fails to be verified. If the signature is successfully verified, the update will be installed on the inactive partition. If the inactive partition already has a software version installed, the update will over-write the previously installed software. Once the new software is installed, the Admin will enter a command in the local console or remote CLI in order to boot off from the inactive partition on which the updated was installed, thus making it the active partition.

Timely Security Updates

As part of providing timely security updates, Gigamon provides customers with a support section on gigamoncp.force.com/gigmoncp/ where they have the ability to submit support issues. This is an HTTPS site that requires user authentication prior to use. Any feedback that necessitates a fix will result in a patch to Gigamon itself so there is no third-party update process to consider when updating the TOE. Any security fixes will be released as new packages in the same manner as any feature. Any implementation flaws are expected to be addressed within 90 days of reporting. Customers are notified of security-related fixes on the Gigamon customer portal.

8.6 TOE Access

8.6.1 **FTA_SSL_EXT.1:**

The TOE is designed to terminate a local session after a specific period of time. The default setting is 15 minutes and it is configurable by an Admin. Once a session has been terminated, the local user must re-authenticate to start a new session.

8.6.2 **FTA_SSL.3:**

The TOE can be configured to terminate remote interactive sessions that are inactive. In the event that the inactivity setting is met while users are logged into the CLI, the TOE tears down the SSH connection. This setting can be configured to 0 or between .25-35791 minutes. The value of 0 means that this setting is disabled and there is no timeout configured which is outside of the evaluated configuration. An Admin user authenticated to the local console or remote CLI may configure this setting for both the local console and remote CLI.

8.6.3 **FTA_SSL.4:**

An Admin is able to terminate their own session by entering the "exit" command when logged into the local console or remote CLI.

8.6.4 **FTA_TAB.1:**

There are two possible ways to authenticate to the TOE: local console or remote CLI. Each of these interfaces has a configurable login banner that is displayed prior to the user authenticating to the TOE.

The banner is configured using the “banner login-local” command for the local console and “banner login-remote” for the remote CLI. The command “banner login” configures the banner for all login methods.

8.7 Trusted Path/Channels

8.7.1 **FTP_ITC.1:**

The TOE connects and sends data to IT entities that reside in the Operational Environment via trusted channels. In the evaluated configuration, the TOE connects with an audit server using SSHv2 to encrypt the audit data that traverses the channel. When remote authentication is configured, the TOE connects to an LDAP Server using TLS v1.2 to send authentication requests for a user attempting to login to the local console or remote CLI. The remote endpoints are authenticated using TLS server certificates and SSH host keys.

In each of these instances, the TOE initiates communication as the client using the cryptographic protocol in the manner described by their respective SFRs. These protocols are used to protect the data traversing the channel from disclosure and/or modification.

8.7.2 FTP_TRP.1/Admin:

The Admin users are required to authenticate to the TOE in order to be able to perform any management functions. By initiating the trusted path via the remote CLI, Admin users can perform management activities remotely. The remote CLI uses SSHv2 which protects the data traversing the channel from disclosure and/or modification.