Adtran's FSP 3000R7 Network Element r22.2.2

Security Target

ST Version: 1.0
Jan 10, 2024

Adtran Networks North America, Inc.
(formerly known as ADVA Optical Networking North America, Inc)
5755 Peachtree Industrial Boulevard
Norcross, Georgia 30092

Prepared By:

Booz | Allen | Hamilton

delivering results that endure

Cyber Assurance Testing Laboratory
1100 West St
Laurel MD 20707

Table of Contents

Table 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.

1.2 ST Identification

ST Title: Adtran's FSP 3000R7 Network Element r22.2.2 Security Target
ST Version: 1.0
ST Publication Date: Jan 10, 2024
ST Author: Booz Allen Hamilton

1.2.1 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.

1.2.2 Terminology

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

Term	Definition
Security Administrator	Represents a person that has authorized access to the TOE to perform configuration and management tasks.
Target of Evaluation (TOE)	A set of software, firmware and/or hardware possibly accompanied by guidance.

Term	Definition
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 (web browser, terminal client, etc.).
TOE Security Function (TSF)	A set consisting of all hardware, software, and firmware of the TOE that must be relied upon for the correct enforcement of the SFRs.
TSF Data	Data for the operation of the TSF upon which the enforcement of the requirements relies.

Table 1: CC Specific Terminology

Term	Definition
CRAFT	The CRAFT is a VT100 terminal style interface that can be accessed at both local and remote administrative access points. The CRAFT is equivalent to the local CLI during local access and the Remote CLI during remote access of this interface. The CRAFT will be referred to as the Local CLI and Remote CLI throughout this document.
Local CLI	When the TOE's command line interface (CLI) is accessed locally with a physical connection to the TOE via Electrical connector type RJ45, 115200 Baud serial port and a VT100 terminal emulator that is compatible with serial communications is referred to as the Local CLI.
NED	The NED is a web interface that can be accessed only through a remote administrative access point. The NED is equivalent to the Web GUI during remote access of this interface. The NED will be referred to as the Web GUI throughout this document.
Security Administrator	The class of TOE administrators that are tasked with managing the TOE's functional and security configuration. Embodies those administrators that have access to the local and remote administrative interfaces.
Remote CLI	The Remote CLI is utilized to perform administrative management functions on the Adtran's FSP 3000R7 at the base operating system level. This interface is accessible over a secure SSH trusted channel from a remote management workstation.
Remote Management Workstation	A standard PC used for remote access to the TOE via either an HTTPS connection or SSH connection.
Terminal	The device that is connected directly to the appliance through the Electrical connector type RJ45 or a serial port. The device will act as a VT100 terminal emulator that is compatible with serial communications used for access to the local CLI.
Web Graphical User Interface (GUI)	The Web GUI is utilized to perform administrative management functions. This interface is accessible over a secure HTTPS trusted channel from a remote management workstation.

Table 2: Customer Specific Terminology

1.2.3 Acronyms

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

Acronym	Definition
CA	Certificate Authority
CC	Common Criteria

Acronym	Definition
CLI	Command-line Interface
DRBG	Deterministic Random Bit Generator
GUI	Graphical User Interface
HTTPS	Hypertext Transfer Protocol Secure
IP	Internet Protocol
IT	Information Technology
NDcPP	Collaborative Protection Profile for Network Devices
NIAP	National Information Assurance Partnership
NTP	Network Time Protocol
OE	Operation Environment
OS	Operating System
PP	Protection Profile
RAM	Random Access Memory
RBAC	Role-Based Access Control
SAR	Security Assurance Requirement
SFR	Security Functional Requirement
SHA	Secure Hash Algorithm
SHS	Secure Hash Standard
SSL	Secure Sockets Layer
SSH	Secure Shell
ST	Security Target
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Function

Table 3: Acronym Definition

1.2.4 Reference

[1] collaborative Protection Profile for Network Devices Version 2.2e 20200323 [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] NIST Special Publication 800-56A Revision 3 Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography, April 2018

[7] FIPS PUB 186-4 Federal Information Processing Standards Publication Digital Signature Standard, July 2013

[8] ISO/IEC 18033-3:2010, Information Technology -- Security techniques -- Encryption algorithms — Part 3: Block ciphers

[9] ISO/IEC 10116:2017, Information Technology -- Security techniques -- Modes of operation for an n-bit block cipher

[10] ISO/IEC 19772:2009, Information Technology – Security techniques – Authenticated encryption
[11] ISO/IEC 10118-3:2004, Information Technology -- Security techniques -- Hash-functions -- Part 3: Dedicated hash-functions
[12] ISO/IEC 9797-2:2011, Information Technology -- Security techniques -- Message Authentication Codes (MACs) -- Part 2: Mechanisms using a dedicated hash-function
[13] ISO/IEC 18031:2011, Information Technology -- Security techniques -- Random bit generation
[14] ISO/IEC 9796-2:2010, Information Technology -- Security techniques -- Digital signature schemes giving message recovery - Part 2: Integer factorization based mechanisms

1.3 TOE Reference

The TOE is Adtran's FSP 3000R7 Network Element operating with software release 22.2.2. In its evaluated configuration, the FSP 3000R7 Network Element is a standalone network device consisting of the Network Control Unit 3 (NCU-3) hardware platform and optional modules for operational network connectivity.

1.4 TOE Overview

The TOE is Adtran's FSP 3000R7 Network Element operating with software release 22.2.2. The TOE, also referred to as the FSP 3000R7 from this point forward, is an optical network management tool. The product is a scalable optical transport solution that is meant to adapt to the bandwidth demands of the network it is deployed in and ensure secure transfer of data across the network.

While the hardware between the three models changes in size to accommodate additional module slots, the processor remains the same. All three models use the same software. The TOE can be deployed as any of the following three model types:

Property	SH1HU	SH7HU	SH9HU
Power	AC/DC/Mix	AC/DC/Mix	AC/DC/Mix
Processor	NCU-3 (NXP QorIQ T-Series T1042E)	NCU-3 (NXP QorIQ T-Series T1042E)	NCU-3 (NXP QorIQ T-Series T1042E)
Size	1 rack unit	7 rack units	9 rack units
Module Slots	2	16	16

Table 4: TOE Models

The following figure depicts the TOE boundary and Operational Environment:

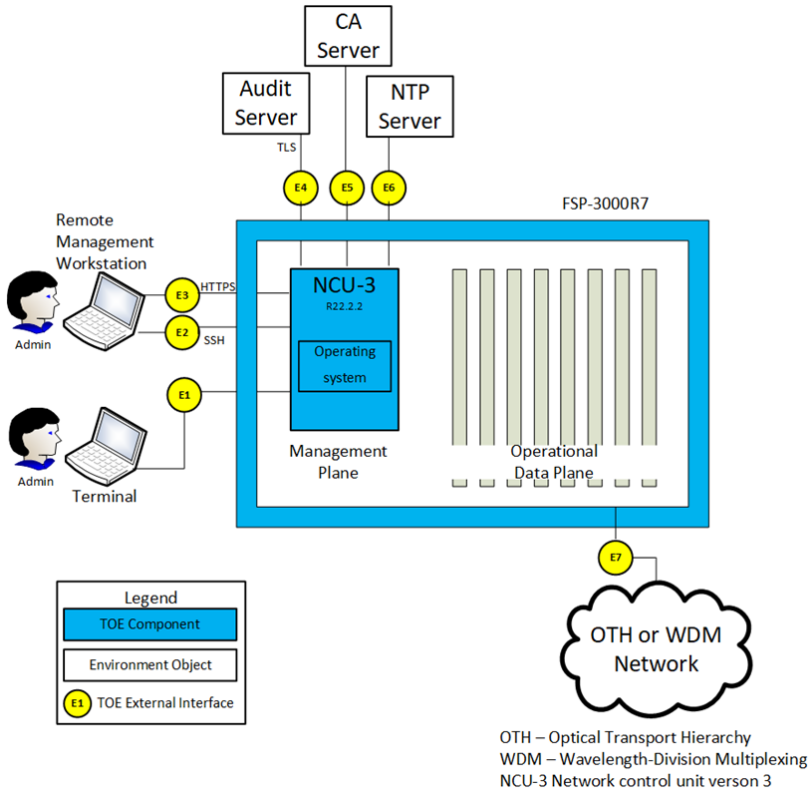

Figure 1: TOE Boundary

The TOE is managed from a dedicated out-of-band network via a Terminal for local administration and a Remote Management Workstation for remote administration.

- E1: A direct local connection from the Terminal to the TOE via a serial or USB port. This connection is used for local administration of the TOE via a CLI.
- E2: A SSHv2 connection from the Remote Management Workstation to the TOE. This connection is used for remote administration of the TOE via a CLI.
- E3: A HTTPS connection from the Remote Management Workstation to the TOE. This connection is used for remote administration of the TOE via a Web GUI.
- E4: A TLS v1.2 trusted channel between the TOE and the external Audit Server used for external audit record storage.
- E5: A connection between the TOE and a Certificate Authority (CRL Distribution Point) used for X.509 certificate verification.
- E6: A connection between the TOE and an NTP server used as its time source. Note: the TOE can also be configured to use an internal clock as its time source.
- E7: FSP 3000R7's connection to the deployed network to provide its optical transport capabilities. While this connection is not part of the evaluated configuration, it is being included for completeness.

While the TOE has the ability to manage the Operational Data Plane, the TOE, in the evaluated configuration, cannot be accessed via the Operational Data Plane. A detailed description of each Operational Environment component is in Table 6 below.

1.5 TOE Type

The TOE type for the FSP 3000R7 is a Network Device. The FSP 3000R7's intended purpose is to manage network traffic within an optical transport network.

The NDcPP defines a network device as "a device that is connected to a network and has an infrastructure role within that network. The TOE may be standalone or distributed, where a distributed TOE is one that requires multiple distinct components to operate as a logical whole in order to fulfil the requirements of this cPP. 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 does not require additional components in order to fulfill its intended purpose and is a standalone appliance. The TOE also consists of both hardware and software. When deployed within a network, the FSP 3000R7 is utilized as a traffic maintenance tool within the network infrastructure and aligns with the requirements of a network device. Therefore, the FSP 3000R7 claims conformance to all NDcPP requirements as claimed in Section 6 of this Security Target.

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 in the evaluated configuration:

TOE Component	Component Description	Hardware Model(s)	Software Version
FSP 3000R7 Network Element	Used by the Security Administrator for optical network transport management	SH1HU, SH7HU, SH9HU	Rel 22.2.2

Table 5: TOE Evaluated Components

2.2 Components and Applications in the Operational Environment

These components and the functionality they provide are outside the scope of evaluation testing but are needed to support the tested functionality of the TOE. The following table lists components and applications that are used in the Operational Environment for the TOE's evaluated configuration.

Component	Definition
Terminal	A terminal is a device that handles the input and display of data when connected to an appliance's serial port. The TOE's CLI can also be accessed locally with a physical connection to the TOE using the Electrical connector type RJ45 or the serial port and must use a VT100 terminal emulator that is compatible with serial communications. Synonymous with the term local console. This OE component is required to support interface E1 as defined in Figure 1 above.
Remote Management Workstation	Any general-purpose computer that is used by an administrator to manage the TOE. For the TOE to be managed remotely the management workstation is required to have:

Component	Definition
	• Supported browser to access the TOE's Web GUI • SSHv2 client installed to access the TOE's CLI The TOE acts as a server for all protocols. TCP communications from the Remote Management Workstation to the TOE is secured using: • SSH for remote access to the CLI • HTTPS for remote access to the Web GUI This OE component is required to support interfaces E2 & E3 as defined in Figure 1 above.
Audit Server	The TOE acts as a TLS client when connected to an Audit Server to send the audit records for remote storage. This OE component is required to support interface E4 as defined in Figure 1 above to send copies of audit data to be stored in a remote location for data redundancy purposes.
Certificate Authority (CA) Server	A server that acts as a trusted issuer of digital certificates and distributes a CRL that identifies revoked certificates. This OE component is required to support interface E5 as defined in Figure 1 above.
NTP Server	The TOE can connect to a NTP Server to maintain accurate timestamps for the TOE and the audit records generated. This OE component is required to support interface E6 as defined in Figure 1 above.
OTH or WDM Network	The OTH or WDM Network represents the optical transport hierarchy and wavelength division multiplexing components. Figure 1 identifies these interfaces as a single interface. The interface to the managed OTH or WDM Network is a separate connection to the enterprise Operational Environment the TOE is managing. There are no SFR's to address the TOE's management of the OTH or WDM Network. Therefore, interface E7 to these components is out of scope for the NDcPP and the present evaluation. This interface and components are included for completeness only.

Table 6: Supporting Components in the Operational Environment

2.3 Excluded from the TOE

The following TOE functionality, components, and/or applications 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 components, applications, and/or functionality that are not installed.

2.3.2 Installed but Requires a Separate License

There are no excluded components, applications, and or functionality that are installed and require a separate license for activation.

2.3.3 Installed But Not Part of the TSF

This section contains functionality that is part of the purchased product but is not part of the TSF relevant functionality that is being evaluated as the TOE based on the Protection Profile.

- The FSP 3000R7 includes a number of capabilities via the Operational Data Plane modules identified in Table 8 below in support of its primary function that are outside the scope of the

claimed Protection Profile. These functions are not part of the TSF because there are no SFRs that apply to them.

2.4 Physical Boundary

The following table outlines the models and their key differentiators that are part of the evaluation.

		FSP 3000R7 Series			
	PROPERTY	SH1HU	SH7HU	SH9HU	Acronym Definitions
Management Plane	Power	AC/DC/Mix	AC/DC/Mix	AC/DC/Mix	NCU-Network Control Unit
	Processor	NCU-3 (NXP QorIQ T-Series T1042E)	NCU-3 (NXP QorIQ T-Series T1042E)	NCU-3 (NXP QorIQ T-Series T1042E)	
	Local Console Connection	RJ45 Jack Serial Connector 1 USB Port	RJ45 Jack Serial Connector 1 USB Port	RJ45 Jack Serial Connector 1 USB Port	
	Management Network Connection	3 RJ45 Ethernet	3 RJ45 Ethernet	3 RJ45 Ethernet	
	Size	1 rack unit	7 rack units	9 rack units	
	Module Slots	2	16	16	
	Commons	FAN/1HU, PSU/1HU-AC, PSU/1HU-DC	FAN/Plug-in, PSU/7HU-AC, PSU/7HU-DC	CEM/9HU, FAN/9HU, PSU/9HU-AC, PSU/9HU-DC	PSU/HU-Power Supply Unit/Housing Unit CEM-Common Equipment Module

Table 7: TOE Model Specifications Management Plane

No NDcPP scoped security functionality is contained within the operational plane. All operational plane functionality is non-SFR supporting, including the encryption modules defined in the table below, and were not tested as part of this evaluation. The following table identifies the plug-in modules that are available for each type of FSP 3000R7 Series model.

		FSP 3000R7 Series			
	PROPERTY	SH1HU	SH7HU	SH9HU	Acronym Definitions
Operational Data Plane	Passive Shelf Accessory	SH1HU/PASSIVE/FT 1 rack unit 4 module slots	SH1HU/PASSIVE/FT 1 rack unit 4 module slots	SH1HU/PASSIVE/FT 1 rack unit 4 module slots	
	Management and Switch Modules	SCU-II, UTM, PSCU, OSCM-PN, HDSCM-PN, OPPM	SCU-II, UTM, PSCU, OSCM-PN, HDSCM-PN, OPPM	SCU-II, OSCM-PN, HDSCM-PN, OPPM	SCU-Shelf Control Unit UTM-Utility Module PSCU-Passive Shelf Control Unit OSCM-Optical Supervisory Channel Module HDSCM-High Density Subshelf Module OPPM-Optical Path Protection Module
	Reconfigurable Optical Modules	4ROADM, MROADM, PSM40, PSM80, 4-OPCM	9ROADM, 4ROADM, MROADM, PSM40, PSM80, 4-OPCM	9ROADM, 4ROADM, MROADM, PSM40, PSM80, 4-OPCM	ROADM-Reconfigurable Optical Add/Drop Module PSM-Power Splitter Module OPCM-Optical Power Control Module

	Optical Amplifiers	GCB, V(L)GC(B), RAMAN, AMP, EDFA, 2EDFA, MA(L)P(B), MTP(B)	GCB, V(L)GC(B), RAMAN, AMP, EDFA, 2EDFA, MA(L)P(B), MTP(B)	GCB, V(L)GC(B), RAMAN, AMP, EDFA, 2EDFA, MA(L)P(B), MTP(B)	EDFA-Erbium Dobed Fiber Amplifier GCB-Gain Controled Balanced V(L)GC(B)-Variable (Low) Gain Controled AMP-EDFA and BWD RAMAN Amplifer Pair MTP(B)-MicroTerminal Pre (Booster) Amplifier MA(L)P(B)-Micro Amplifer (Low) Pre (Booster) Amplifiers
	Access Modules	2WCA, 5WCA, 6WCA	2WCA, 5WCA, 6WCA	2WCA, 5WCA, 6WCA	WCA-Wavelength Converter Module Access
	Core 100G Modules	WCC, 4TCC, 10TCC		WCC, 4TCC, 10TCC	WCC-Wavelength Channel Module Core TCC-TDM Channel Module Core
	Core <100G Modules	2(16)TCC, 2(4)WCC	2(16)TCC, 2(4)WCC	2(16)TCC, 2(4)WCC	
	Enterprise 100G Modules	10TCE		10TCE	TCE-TDM Channel Module Enterprise
	Enterprise <100G Modules	9TCE	9TCE	9TCE	
	Encryption Modules	9TCE+AES, 10TCE+AES, WCC+AES	9TCE+AES	9TCE+AES, 10TCE+AES, WCC+AES	+AES-FIPS 140-2 Encryption
	Dispersion Compensation Modules	DCF, DCG	DCF, DCG	DCF, DCG	DCF-Dispersion Compensation Fiber Modules DGG-Dispersion Compensation Fiber-Bragg_Gratings
	Passive Filter Modules	OSFM(A),(x)PSM, (x)PM, (x)PSM(x), ILM, (x)CSM	OSFM(A),(x)PSM, (x)PM, (x)PSM(x), ILM, (x)CSM	OSFM(A),(x)PSM, (x)PM, (x)PSM(x), ILM, (x)CSM	OSFM(A)-Optical Supervisory Filter Module (integrated ALM/OTC Coupler) PSM-Port Splitter Module PM-Protection Splitter Module ILM-Interleaver Module for even/odd channels CSM-Channel Splitter Modules

Table 8: TOE Model Specifications Operational Plane

2.5 Logical Boundary

The TOE is comprised of the following security features that have been scoped by the protection profile:

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

2.5.1 Security Audit

The TOE contains mechanisms to generate audit data to record predefined events on the TOE. The TOE has the ability to store audit logs locally and will free up audit storage space by deleting archived files in a First in First out (FIFO) fashion. The Security Administrator can configure the forwarding of events to an

external Audit Server. In the evaluated configuration, the audit data is securely transmitted to the Audit Server using a TLS v1.2 communication channel.

2.5.2 Cryptographic Support

The TOE provides cryptography in support of SSH and TLS (v1.2) trusted communications. OpenSSL is used for all TLS and SSH communications. The TOE immediately destroys keys when no longer used. The following table identifies the cryptographic services:

SFR	OpenSSL Implementation	
FCS_CKM.1	ECC schemes using NIST curves P-384 following FIPS PUB 186-4	CAVP #A4284
	FFC using safe-prime groups NIST Special Publication 800-56A Revision 3 and RFC 3526.	N/A
FCS_CKM.2	Elliptic curve-based key establishment per NIST Special Publication 800-56A Revision 3	CAVP #A4284
	FFC using safe-prime groups NIST Special Publication 800-56A Revision 3 and RFC 3526.	N/A
FCS_COP.1/DataEncryption	AES CTR 256 bits AES GCM 256 bits	CAVP #A4284
FCS_COP.1/SigGen	ECDSA FIPS 186-4 Signature Services 384 bits	CAVP #A4284
FCS_COP.1/Hash	SHA-384 and SHA-512	CAVP #A4284
FCS_COP.1/KeyedHash	HMAC-384	CAVP #A4284
FCS_RBG_EXT.1	CTR DRBG (AES-256)	CAVP #A4284

Table 9: Cryptographic Services

2.5.3 Identification and Authentication

The TOE enforces the use of X.509 certificates to support authentication for all TLS connections. The TOE provides a password-based authentication mechanism for users to access the local CLI, remote CLI and Web GUI. The TSF will lock a user's account from remote access after a configurable number of failed login attempts has been reached. Feedback from password entry is always obscured during local authentication. The only function available to an unauthenticated user is the ability to acknowledge a warning banner.

2.5.4 Security Management

The TOE uses role-based access control to prevent unauthorized management of and access to TSF data. The TOE maintains the role of Security Administrator which is able to administer the TOE locally and remotely.

2.5.5 Protection of the TSF

The TOE ensures the security and integrity of all data that is stored locally and accessed remotely. Passwords are not stored in plaintext. A Security Administrator has the ability to query the currently executing version of the TOE software and is required to manually initiate the update process. Prior to installation, the TOE automatically verifies the X.509 certificate used to sign the software update. In the evaluation configuration, if the certificate is found to be invalid for any reason or is missing, the update is

not installed. The TOE implements a self-testing mechanism that is automatically executed during the initial start-up to verify the correct operation of the TOE and cryptographic functions. The TOE provides its own time either via its administratively configurable internal clock or via a connection to an NTP Server.

2.5.6 TOE Access

The TOE displays a configurable warning banner prior to user authentication. Users have the ability to terminate their own interactive session. Local and remote sessions are automatically terminated after the administrator configured inactivity time limit is reached.

2.5.7 Trusted Path/Channels

Users can access the CLI for administration functions locally via a physical connection to the TOE or remotely via a SSH connection where the TOE acts a SSH Server. Users can also access the Web GUI for remote administrative functionality via a HTTPS connection where the TOE acts as a HTTPS/TLS server.

The TOE acts as a TLS client to initiate the secure channel to an external Audit Server.

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 Jan 10, 2024.

3.3 CC Part 3 Conformance Claims

This ST and Target of Evaluation (TOE) is Part 3 conformant to include all applicable NIAP and International interpretations through Jan 10, 2024.

3.4 PP Claims

This ST claims exact conformance to the following Protection Profiles:

- Collaborative Protection Profile for Network Devices Version 2.2e (NDcPP), March 23, 2020

3.5 Package Claims

The TOE claims exact compliance to the Collaborative Protection Profile for Network Devices Version 2.2e, 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_HTTPS_EXT.1
- FCS_NTP_EXT.1
- FCS_SSHS_EXT.1

- FCS_TLSC_EXT.1
- FCS_TLSS_EXT.1
- FIA_X509_EXT.1/Rev
- FIA_X509_EXT.2
- FIA_X509_EXT.3
- FMT_MTD.1/CryptoKeys
- FPT_TUD_EXT.2

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

- FCS_TLSS_EXT.2

The PP specifically indicates optional SFRs as allowable options. Therefore, the notion of exact conformance is not violated when not all optional SFRs are claimed. The PP 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 version 2.2e.

3.7 Technical Decisions

Technical Decisions that affected the SFR wording have been annotated with a Footnote.

The following NDcPP Technical Decisions apply to the TOE because SFR wording, application notes, or assurance activities were modified for SFRs claimed by the TOE:

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 v2.1, ND SD v2.2		X			AA: Testing Update.
TD0536	NIT Technical Decision for Update Verification Inconsistency	AGD_OPE.1, ND SDv2.1, 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 ST.
TD0546	NIT Technical Decision for DTLS - clarification of Application Note 63	FCS_DTLSC_EXT.1.1			X	X	N/A: SFR not claimed
TD0547	NIT Technical Decision for Clarification on developer disclosure of AVA_VAN	ND SDv2.1 ND SDv2.2, AVA_VAN.1		X			Clarification of AVA_VAN No ST updates required.

TD #	Title	References	Changes			Analysis to this evaluation	
			SFR	AA	Notes	NA	Reason
TD0555	NIT Technical Decision for RFC Reference incorrect in TLSS Test	NDSDv2.2, FCS_TLSS_EXT.1.4, Test 3		X			AA: Test clarification no wording change No ST updates required.
TD0556	NIT Technical Decision for RFC 5077 question	NDSDv2.2, FCS_TLSS_EXT.1.4, Test 3		X		X	N/A: Test for renegotiation does not apply.
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.
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	AA: TSS, AGD, ATE Neither session tickets nor resumption is 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		AA:TSS, Test See Footnote 2
TD0581	NIT Technical Decision for Elliptic curve-based key establishment and NIST SP 800-56Arev3	FCS_CKM.2	X				SFR word changes to update Revision number to 3. See Footnote 2

TD #	Title	References	Changes			Analysis to this evaluation	
			SFR	AA	Notes	NA	Reason
TD0591	NIT Technical Decision for Virtual TOEs and hypervisors	A.LIMITED_FUNCTIONALITY, ACRONYMS			X		Assumption wording change. See Footnote 1.
TD0592	NIT Technical Decision for Local Storage of Audit Records	FAU_STG				X	Clarification of PP text. N/A: SFR not claimed
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, Testing Update. Footnote 3
TD0632	NIT Technical Decision for Consistency with Time Data for vNDs	ND SD2.2, FPT_STM_EXT.1.2	X			X	N/A: TOE is not a vND
TD0635	NIT Technical Decision for TLS Server and Key Agreement Parameters	FCS_TLSS_EXT.1.3, NDSD v2.2		X			AA: TSS update
TD0636	NIT Technical Decision for Clarification of Public Key User Authentication for SSH	ND SD2.2, FCS_SSHC_EXT.1	X	X	X	X	N/A: Not claiming SSH Client functionality
TD0638	NIT Technical Decision for Key Pair Generation for Authentication	NDSDv2.2, FCS_CKM.1			X		Requires selection of all key generation schemes needed for FTP_ITC.1, FTP_TRP.1/Admin, FTP_TRP.1/Join, and FPT_ITT.1 in this SFR
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		Clarification; no changes to AA or ST required.
TD0670	NIT Technical Decision for Mutual and Non-Mutual Auth TLSC Testing	ND SD2.2, FCS_TLSC_EXT.2.1		X		X	AA: Testing Updated Not claiming mutual auth support for client.
TD0738	NIT Technical Decision for Link to Allowed-With List	Chapter 2			X		PP claimed but note change has no impact on ST.
TD0790	NIT Technical Decision: Clarification Required for testing IPv6	FCS_DTLSC_EXT.1.2, FCS_TLSC_EXT1.2, CPP_ND_V2.2-SD		X			AA: Test Not claiming IPv6 however, test procedure wording has been updated
TD0792	0792 – NIT Technical Decision: FIA_PMG_EXT.1 - TSS EA not in line with SFR	FIA_PMG_EXT.1, CPP_ND_V2.2-SD		X			AA:TSS
TD0800	0800 – Updated NIT Technical Decision for IPsec IKE/SA Lifetimes Tolerance	FCS_IPSEC_EXT.1.7, FCS_IPSEC_EXT.1.8, CPP_ND_V2.2-SD		X		X	AA:AGD, Test N/A: SFRs not claimed

Table 10: Technical Decisions

3.8 Conformance Claim Rationale

Section 1.2 of the NDcPP states: The NDcPP defines a network device as "a device that is connected to a network and has an infrastructure role within that network. The TOE may be standalone or distributed, where a distributed TOE is one that requires multiple distinct components to operate as a logical whole in order to fulfil the requirements of this cPP…" Additionally, the NDcPP says that example devices that fit this definition include "physical and virtualised routers, firewalls, VPN gateways, IDSs, and switches."

The FSP 3000R7 is a scalable optical transport solution that is meant to adapt to the bandwidth demands of the network it is deployed in and ensure secure transfer of data across the network. The TOE is a standalone network device, composed of hardware and software, and provides an infrastructure role for the network it is deployed in. Therefore, this conformance claim is appropriate, and the TOE type is justified.

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

Threat	Threat Definition
	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 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 11: 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 12: TOE Organization 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). 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 13: TOE Assumptions

[1] TD0591

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 14: TOE 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. The associated mappings of assumptions to environmental objectives, SFRs to TOE objectives, and OSPs and objectives to threats are therefore identical to the mappings that are specified in the claimed Protection Profile.

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

There are no extended Security Assurance Requirements in this ST.

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.
- **Selection:** allows the specification of one or more elements from a list. Indicated with underlined 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. "/LocSpace" for an SFR that relates to local storage space

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_GEN.1	Audit Data Generation
	FAU_GEN.2	User identity association
	FAU_STG_EXT.1	Protected Audit Event Storage
Cryptographic Support	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_NTP_EXT.1	NTP Protocol
	FCS_RBG_EXT.1	Random Bit Generation
	FCS_HTTPS_EXT.1	HTTPS Protocol.
	FCS_SSHS_EXT.1	SSH Server Protocol
	FCS_TLSC_EXT.1	TLS Client Protocol Without Mutual Authentication
	FCS_TLSS_EXT.1	TLS Server Protocol Without Mutual Authentication
	FCS_TLSS_EXT.2	TLS Server Support for Mutual Authentication
Identification and Authentication	FIA_AFL.1	Authentication Failure Management
	FIA_PMG_EXT.1	Password Management
	FIA_UAU.7	Protected Authentication Feedback
	FIA_UAU_EXT.2	Password-based Authentication Mechanism
	FIA_UIA_EXT.1	User Identification and Authentication

Class Name	Component Identification	Component Name
	FIA_X509_EXT.1/Rev	X.509 Certificate Validation
	FIA_X509_EXT.2	X509 Certificate Authentication
	FIA_X509_EXT.3	X509 Certificate Requests
Security Management	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_APW_EXT.1	Protection of Administrator Passwords
	FPT_SKP_EXT.1	Protection of TSF Data (for reading of all symmetric keys)
	FPT_STM_EXT.1	Reliable Time Stamps
	FPT_TST_EXT.1	TSF Testing
	FPT_TUD_EXT.1	Trusted Update
	FPT_TUD_EXT.2	Trusted Update Based on Certificate
TOE Access	FTA_SSL_EXT.1	TSF-initiated Session Locking
	FTA_SSL.3	TSF-initiated Termination
	FTA_SSL.4	User-initiated Termination
	FTA_TAB.1	Default TOE Access Banner
Trusted Path /Channels	FTP_ITC.1	Inter-TSF trusted channel
	FTP_TRP.1/Admin	Trusted Path

Table 15: 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 **16**.

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 **16**.

Requirement	Auditable Event(s)	Additional Audit Record Contents
FAU_GEN.1	None.	None.
FAU_GEN.2	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_NTP_EXT.1	• Configuration of a new time server • Removal of configured time server	Identity if new/removed time server
FCS_RBG_EXT.1	None.	None.
FCS_HTTPS_EXT.1	Failure to establish a HTTPS 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
FCS_TLSS_EXT.1	Failure to establish a TLS session	Reason for failure
FCS_TLSS_EXT.2	Failure to authenticate the client	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.7	None.	None.
FIA_UAU_EXT.2	All use of the identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UIA_EXT.1	All use of the 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 • Identification of certificates added, replaced or removed as trust anchor in the TOE's trust store
FIA_X509_EXT.2	None.	None.
FIA_X509_EXT.3	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.

Requirement	Auditable Event(s)	Additional Audit Record Contents
FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure)	None.
FPT_TUD_EXT.2	Failure of update	Reason for failure (including identifier of invalid certificate)
FTA_SSL_EXT.1 (if "terminate the session" is selected)	The termination of a local session by the session locking mechanism.	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_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.	Identification of the claimed user identity.

Table 16: 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_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 [[*rotate audit log files on a First-in-First-out (FIFO) basis according to the following rule:*

- *delete oldest archived audit log file*
- *archive current audit log file (close, compress, and rename file)*
- *open a new audit log file for receiving current audit records*

]]

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-384] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4;
- FFC Schemes using 'safe-prime' groups that meet the following: "NIST Special Publication 800-56A Revision 3, Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" and [RFC 3526].

].

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

].

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 [
    - instructs a part of the TSF to destroy the abstraction that represents the key]

that meets the following: No Standard.

[2] TD0580 and 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 [CTR, GCM] mode and cryptographic key sizes [256 bits] that meet the following: AES as specified in ISO 18033-3, [CTR 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 *[384 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-384]; 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-384, SHA-512] and message digest sizes [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-384] and cryptographic key sizes [*384 bits*] and message digest sizes [384] bits that meet the following: ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2".

6.3.2.8 *FCS_NTP_EXT.1 NTP Protocol*

FCS_NTP_EXT.1.1

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

FCS_NTP_EXT.1.2

The TSF shall update its system time using [

- Authentication using [SHA384] as the message digest algorithm(s);

].

FCS_NTP_EXT.1.3

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

FCS_NTP_EXT.1.4

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

6.3.2.9 *FCS_RBG_EXT.1 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 [[*1*] software-based noise source, [*1*] platform-based noise source] with a minimum of [256 bits] of entropy at least equal to the greatest security strength, according to ISO/IEC 18031:2011 Table C.1 "Security Strength Table for Hash Functions", of the keys and hashes that it will generate.

6.3.2.10 *FCS_HTTPS_EXT.1 HTTPS Protocol*

FCS_HTTPS_EXT.1.1

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

FCS_HTTPS_EXT.1.2

The TSF shall implement HTTPS using TLS.

FCS_HTTPS_EXT.1.3

If a peer certificate is presented, the TSF shall [not establish the connection] if the peer certificate is deemed invalid.

6.3.2.11 *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, [5647, 5656, 8268].

FCS_SSHS_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, [password-based].

FCS_SSHS_EXT.1.3

[3] TD0631

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*:* [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 [implicit] as its MAC algorithm(s) and rejects all other MAC algorithm(s).

FCS_SSHS_EXT.1.7

The TSF shall ensure that [diffie-hellman-group15-sha512] and [ecdh-sha2-nistp384] 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.

6.3.2.12 *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_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 [IPv4 address in CN or SAN 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: [secp384r1] and no other curves/groups] in the Client Hello.

6.3.2.13 *FCS_TLSS_EXT.1 TLS Server Protocol Without Mutual Authentication*

FCS_TLSS_EXT.1.1

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

- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289

and no other ciphersuites.

FCS_TLSS_EXT.1.2

The TSF shall deny connections from clients requesting SSL 2.0, SSL 3.0, TLS 1.0, and [TLS 1.1].

FCS_TLSS_EXT.1.3

The TSF shall perform key establishment for TLS using [ECDHE curves [secp384r1] and no other curves].

FCS_TLSS_EXT.1.4

The TSF shall support [no session resumption or session tickets].

6.3.2.14 *FCS_TLSS_EXT.2 TLS Server Protocol for Mutual Authentication*

FCS_TLSS_EXT.2.1

The TSF shall support TLS communication with mutual authentication of TLS clients using X.509v3 certificates.

FCS_TLSS_EXT.2.2

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

- Not implement any administrator override mechanism

].

FCS_TLSS_EXT.2.3

The TSF shall not establish a trusted channel if the identifier contained in a certificate does not match an expected identifier for the client. If the identifier is a Fully Qualified Domain Name (FQDN), then the TSF shall match the identifiers according to RFC 6125, otherwise the TSF shall parse the identifier from the certificate and match the identifier against the expected identifier of the client as described in the TSS.

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 [*1-10*] 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 remote session using any authentication method that involves a password until [*a manual unlock of the account*] is taken by an Administrator; prevent the offending Administrator from successfully establishing remote session using any authentication method that involves a password until an Administrator defined time period has elapsed].

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 [*15*] and [*128*] characters.

6.3.3.3 *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.

6.3.3.4 *FIA_UAU_EXT.2* *Password-based Authentication Mechanism*

FIA_UAU_EXT.2.1

The TSF shall provide a local [password-based] authentication mechanism to perform local administrative user authentication.

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

FIA_X509_EXT.2.2

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

6.3.3.8 *FIA_X509_EXT.3 X.509 Certificate Requests*

FIA_X509_EXT.3.1

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

FIA_X509_EXT.3.2

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

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[4]

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 re-enable an Administrator account;
  - Ability to set the time which is used for time-stamps;
  - Ability to configure NTP;
  - Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors;
  - Ability to import X.509v3 certificates to the TOE's trust store;
  - Ability to manage the trusted public keys database].

6.3.4.5 *FMT_SMR.2 Restrictions on Security Roles*

[4] TD0631

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.

6.3.5.2 *FPT_SKP_EXT.1 Protection of TSF Data (for reading of all symmetric 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, synchronize time with an NTP server].

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, validation of cryptographic functions, file system integrity*].

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 [X.509 certificate, published hash] prior to installing those updates.

6.3.5.6 *FPT_TUD_EXT.2 Trusted Update Based on Certificates*

FPT_TUD_EXT.2.1 The TSF shall check the validity of the code signing certificate before installing each update.

FPT_TUD_EXT.2.2 If revocation information is not available for a certificate in the trust chain that is not a trusted certificate designated as a trust anchor, the TSF shall [not install the update].

FPT_TUD_EXT.2.3 If the certificate is deemed invalid because the certificate has expired, the TSF shall [not accept the certificate].

FPT_TUD_EXT.2.4 If the certificate is deemed invalid for reasons other than expiration or revocation information being unavailable, the TSF shall not install the update.

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 Default 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 [TLS] to provide a trusted communication channel between itself and authorized IT entities supporting the following capabilities: audit server, [no other capabilities] that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from disclosure and detection of modification of the channel data.

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 [*exporting of audit records to audit server*].

6.3.7.2 *FTP_TRP.1/Admin Trusted Path*

FTP_TRP.1.1/Admin

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

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 PPs against which exact conformance is claimed and a subset of the optional SFRs. All hierarchical relationships, dependencies, and unfulfilled dependency rationales in the ST are considered to be identical to those that are defined in 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. In the case of entropy analysis, the TSS is used in conjunction with required supplementary information on Entropy.

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 Support

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 and FAU GEN.2

The TOE has the mechanisms to automatically generate audit records based on the behavior and events that occurs within the TSF. The audit functionality begins automatically with startup of the TOE. The TOE generates audit records for all administrative functions and events including Login/Logout, security related changes, resetting of passwords, and certificate management. Additionally, Table 16 identifies the audit records that are inclusive to the PP evaluation scoping. The TOE records the date and time, type of event, subject identity (identity of the user associated with each audited event that occurred due to a user action), and the outcome within the audit record. For a full list of the audit events samples that are generated by the TOE, please refer to the Supplemental Administrative Guidance Document (AGD).

The following is an example audit record produced for importing a certificate for use of pub key encryption (Timestamp, USER: Admin, Event: ADD_KEY with details including: purpose, key algorithm, length and fingerprint of certificate.

```
192.168.1.75 WDM[5841] 19342 2023-08-30T18:09:27.44Z ADD_KEY: USER=ADMIN,
ACCESS=LOCAL, KEY-ALGORITHM=ECDSA, KEY-LENGTH=384, FINGERPRINT-
STR=SHA256:RPZzbFJBbvvMPqqqgyfZTNCSGF35EuiHjMcBxC8qaas, ALIAS=catl@DESKTOP-
7P0THJ7
```

8.1.2 FAU_STG_EXT.1

The TSF allows a Security Administrator to configure the near real-time forwarding of the audit trail to an external Audit Server in the Operational Environment. The TOE is a standalone appliance responsible for storing and sending its own generated audit records to the external Audit Server. Once configured, generated audit data is first saved locally on the TOE. The TOE then securely transmits audit data via a TLS channel to the external Audit Server in the Operational Environment without administrator intervention. During a connection outage to the Audit Server, the TOE continues to save audit data locally. Once the connection to the Audit Server is re-established, the TOE automatically starts forwarding new audit records. The TOE does not forward the records created during the outage.

The TOE compresses audit log files in order to reduce the storage footprint of the audit records within each log's respective storage location. When the current audit log file reaches its maximum file size or number of entries, the TOE will rotate audit log files in the following manner:

- If the maximum number of archived audit log files exists: delete oldest archived audit log file in order to maintain the maximum number of archived audit log files (FIFO methodology)
- Archive current audit log file (close, compress, and rename file)
- Open a new audit log file for receiving current audit records

Audit log files are archived according to the type of audit log following the below archiving rules:

Audit Log File Type	Min Audit Log File Size to Trigger Archiving	Max Number of Audit Record Entries per Log File (M)	Max Number of Archived Audit Log Files (N)	Max Audit Log File Storage Space (Current + Archived)
Condition Log (Event)	-	100	19	2 MB (*)
Database Change Log	-	50	19	2 MB (*)
Security Log (System)	≥80 KB	-	9	800KB

Table 17: Audit Log Archiving Rules

(*) the Condition and Database Change logs are kept in **one shared** SRAM 2MB-sized partition.

The audit records may be viewed using any of the Security Administrator interfaces. There is no access to delete or modify audit records through the Web GUI. However, the audit log files can be accessed at the OS level by a Security Administrator that has the ability to escalate to root privileges, using the sudo command, to make authorized file deletions or modifications.

8.2 Cryptographic Support

The TOE implements the OpenSSL cryptographic library. The OpenSSL library include algorithms that are certified under the following consolidated CAVP certificates:

a) OpenSSL library under CAVP Certificate #4284

The following tables contain the CAVP algorithm certificates for the cryptographic library implemented in the TOE:

SFR	Algorithm/Protocol	OpenSSL CAVP Cert #
FCS_CKM.1	ECC schemes using NIST curves P-384 following FIPS PUB 186-4	ECDSA CAVP Certificate #A4284
	FFC using safe-prime groups NIST Special Publication 800-56A Revision 3 and RFC 3526.	N/A
FCS_CKM.2	Elliptic curve-based key establishment per NIST Special Publication 800-56A Revision 3	KAS ECC CAVP Certificate #A4284
	FFC using safe-prime NIST Special Publication 800-56A Revision 3 and groups listed in RFC 3526.	N/A
FCS_COP.1/DataEncryption	AES CTR 256 bits AES GCM 256 bits	AES CAVP Certificate #A4284
FCS_COP.1/SigGen	ECDSA FIPS 186-4 Signature Services 384 bits	ECDSA SigGen and Sig Ver CAVP Certificate #A4284
FCS_COP.1/Hash	SHA-384 and SHA-512	SHS

		CAVP Certificate #A4284
FCS_COP.1/KeyedHash	HMAC-384	HMAC CAVP Certificate #A4284
FCS_RBG_EXT.1	CTR DRBG (AES-256)	CTR DRBG CAVP Certificate #A4284

Table 18: Cryptographic Algorithm Table for OpenSSL

8.2.1 FCS_CKM.1

The TOE implements a FIPS PUB 186-4 conformant ECC key generation mechanism for establishing TLS connections. Specifically, the TOE's implementation of ECC key generation complies with FIPS 186-4 (Digital Signature Standard (DSS) Appendix B.4) supporting a 384-bit key size.

The TOE also implements FFC key generation mechanism using safe prime groups for establishing SSH connections. The TOE's implementation of FFC key generation complies with NIST Special Publication 800-56A Revision 3 and RFC 3526 supporting a key size of 1024 bits.

OpenSSL provides the key generation services for ECC and FFC certificate creation. See Table 18 Cryptographic Algorithm Table for certification numbers.

8.2.2 FCS_CKM.2

The TOE implements Elliptic curve-based key establishment, conformant to NIST Special Publication 800-56A Revision 3 in support of the TOE's TLS client and server services (FCS_TLSC_EXT.1 and FCS_TLSS_EXT.1). The TOE complies with NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" and all subsections regarding Elliptic curve-based key pair generation and key establishment. The Elliptic curve-based key establishment is used for TLS communications for remote administration via the Web GUI and exporting audit data to the Audit Server.

The TOE implements FFC based key establishment using safe prime groups in support of the TOE's SSH server service (FCS_SSHS_EXT.1). The TOE complies with NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography", and groups listed in RFC 3526. The FFC based key establishment is utilized by the TOE for SSH communications between itself and the Remote Administrative Workstation.

8.2.3 FCS_CKM.4

The following table describes what keys are used, the keys origin, where the keys are stored, and how the keys are destroyed. There are no known instances where key destruction does not happen as defined.

Name	Origin	Store	Zeroization / Destruction
Diffie-Hellman Shared Secret	SSH Server / client applications	RAM	Destroyed by a single direct overwrite consisting of zeroes (0x00). The key is zeroized immediately after it is no longer needed and when the TOE is

Name	Origin	Store	Zeroization / Destruction
			shutdown or reinitialized. Automatically zeroized after DH exchange. From openSSH: sshbuf_free() BN_clear_free()
Diffie-Hellman private exponent	SSH Server / client applications	RAM	Destroyed by a single direct overwrite consisting of zeroes (0x00). The key is zeroized immediately after it is no longer needed and when the TOE is shutdown or reinitialized. Automatically zeroized after DH exchange From openSSH: DH_free()
SSH session key	SSH Server / client applications	RAM	Destroyed by a single direct overwrite consisting of zeroes (0x00). The key is zeroized immediately after it is no longer needed and when the TOE is shutdown or reinitialized. Automatically zeroized after SSH session is terminated. From openSSH: sshbuf_free()
SSH Server Host Private Key	Generated on platform during initial setup of device.	Filesystem	Filesystem: Generation of a new key will only be accomplished during a regeneration of the product's SSH Server key. As part of this process all old files would be overwritten effectively destroying the abstraction that represented the key. From TOE: Server process uses [rm -f] to destroy a private key file
TLS Server Host Certificate Private Key	Generated on platform (OpenSSL) during initial setup or imported after installation. OpenSSL TLS Communications for Audit Server and AD	RAM and Filesystem	RAM: The Server Certificate's private key is destroyed by a single direct overwrite consisting of zeroes (0x00). The key is zeroized immediately after it is no longer needed and when the TOE is shutdown or reinitialized. From nginx: EVP_PKEY_free() Filesystem: Private key is deleted with the generation of new certificates that overwrites, import of new certificates that

Name	Origin	Store	Zeroization / Destruction
			overwrites, or when certificates are removed. As part of this process all old files would be deleted effectively destroying the abstraction that represented the key. From TOE: SecuServer process uses rm – frv command to destroy a private key file

Table 19: Crypto Key Destruction Table

8.2.4 FCS_COP.1/DataEncryption

The TOE performs encryption and decryption using the AES algorithm in CTR and GCM modes with key sizes of 256 bits. The AES algorithm meets ISO 18033-3, CTR meets ISO 10116, and GCM meets ISO 19772. The TOE's AES implementation is validated under CAVP. See Table 18 Cryptographic Algorithm Table for certification numbers.

OpenSSL provides the encryption and decryption algorithms to support:

- SSH and TLS communication: AES-GCM-256
- CTR DRBG: AES-CTR-256

8.2.5 FCS_COP.1/SigGen

The TOE performs digital signature services generation and verification in accordance with Elliptic Curve Digital Signature Algorithm (ECDSA) with key sizes (modulus) 384 bits. The ECDSA schemes are in accordance with FIPS PUB 186-4, "Digital Signature Standard (DSS)", section 6 and appendix D, Implementing "NIST curves" [P-384]; ISO/IEC 14888-3, Section 6.4. The TOE's ECDSA implementation is validated under CAVP. See Table 18 Cryptographic Algorithm Table for certification numbers. Digital signature services are utilized by the TOE for the following events:

- mutual authentication of Web Browser;
- connecting to a remote syslog collector;
- checking signature of a software updates

8.2.6 FCS_COP.1/Hash

The TOE provides cryptographic hashing services using SHA-384 and SHA-512 as specified in ISO/IEC 10118-3:2004 (FIPS PUB 180-4) using the OpenSSL library. The TOE's SHS implementation is validated under CAVP. See Table 18 Cryptographic Algorithm Table for certification numbers. The TSF uses hashing services the following functions:

- SHA-384 for TLS (FCS_TLSC_EXT.1/ FCS_TLSS_EXT.1)
- SHA-384 for TLS NIST curves (FCS_TLSC_EXT.1/ FCS_TLSS_EXT.1)
- SHA-384 for HMAC (FCS_COP.1/KeyedHash)
- SHA-384 for software integrity check (FPT_TST_EXT.1)
- SHA-384 for NTP timestamp verification (FCS_NTP_EXT.1)
- SHA-384 for code signing certificate's signature hash algorithm (FPT_TUD_EXT.1)
- SHA-512 for password hashing (FPT_APW_EXT.1)

8.2.7 **FCS_COP.1/KeyedHash**

The TOE provides keyed-hashing message authentication services that meet ISO/IEC 9797-2:2011 (FIPS PUB 198-1, and FIPS PUB 180-4), Section 7 "MAC Algorithm 2". The OpenSSL library utilized for the TOE supports the following:

- HMAC-SHA-384 [key-size: 384 bits, digest size: 384 bits, block size: 1024 bits, MAC lengths: 384 bits] for TLS communication support.

The TOE's HMAC implementation is validated under CAVP. See Table 18 Cryptographic Algorithm Table for certification numbers.

8.2.8 **FCS_NTP_EXT.1**

In its evaluated configuration, the TOE's time source can be configured via its internal clock or an NTP Server. When the TOE is configured to use an NTP Server, the TOE utilizes NTP v4 applied in accordance with RFC 5905 and no other version of NTP. The system time is updated via NTP client-server authentication. The TOE uses SHA-384 message digest algorithm to verify the authenticity of the timestamp which ensures reliability. The TOE supports a maximum of 3 NTP servers. The TOE will not update NTP timestamp from broadcast and/or multicast addresses.

8.2.9 **FCS_RBG_EXT.1**

The TOE's implementation of OpenSSL uses a counter mode random bit generator (CTR DRBG) and complies with ISO/IEC 18031:2011. There is no ability to specify the use of an alternative DRBG. The DRBG used by the TOE uses 1 platform-based and 1 software-based noise source as stated in the proprietary entropy specification. The DRBG is seeded with a minimum of 256-bit security strength.

The TOE relies on kernel modules (software) to gather and output entropy for the TOE's random requirements. Additionally, the TOE uses a hardware source to produce entropy to fill the entropy pool quicker during the boot process. This hardware source is not used during operational runtime. The entropy pools are protected by being in kernel memory and are not accessible from user space. The entropy source is described in greater detail in the proprietary Entropy Assessment Report.

The TOE's DRBG implementation is validated under CAVP. See Table 18 Cryptographic Algorithm Table for certification numbers.

8.2.10 **FCS_HTTPS_EXT.1**

The TOE implements HTTPS for remote administration over a Web GUI. The HTTPS implementation conforms to RFC 2818 and uses the TLS server implementations that cover the functionality specified in FCS_TLSS_EXT.1 and FCS_TLSS_EXT.2 (mutual authentication). The TOE will validate the presented certificate immediately upon receipt as part of the TLS handshaking in accordance with FIA_X509_EXT.1/Rev. The certificate will be validated first and then the TSF will perform a revocation status check if the certificate is deemed valid. If the certificate is deemed invalid for any reason the TSF will immediately terminate the connection.

8.2.11 **FCS_SSHS_EXT.1**

The TOE acts as an SSHv2 server for Remote CLI sessions that complies with RFCs 4251, 4252, 4253, 4254, 5647, 5656, and 8268. The SSH secure channel is utilized by the Security Administrator for remote

administration of the TOE through the Remote CLI. The TOE's implementation of SSH supports public key-based and password-based user authentication. If a public key is presented for user authentication, the TOE will verify that the SSH client's presented public key matches one that is stored within the SSH server's authorized keys database. In the case of password-based authentication attempt, the presented user credentials are verified using the TOE's native password authentication mechanism.

The SSH implementation will detect all large packets greater than 32,768 bytes and drop accordingly. Additionally, the TSF enforces the connection to be rekeyed after no longer than one hour, and no more than one gigabyte of transmitted data, whichever threshold is reached first. The SSH rekey time and size threshold parameters are not administratively configurable.

The TOE's implementation of SSHv2 only supports:

- aes256-gcm@openssh.com for its only encryption algorithms
- ecdsa-sha2-nistp384 as its only public key algorithm (user and host)
- MAC algorithm is implicit due to the selection of aes256-gcm@openssh.com for encryption algorithm
- diffie-hellman-group15-sha512 and ecdh-sha2-nistp384 for key exchange method

OpenSSH provides all cryptographic support required for SSH communication.

8.2.12 **FCS_TLSC_EXT.1**

The TOE implements OpenSSL to provide the cryptographic support for key establishment and encryption for these TLS channels when the TOE acts as client. The TOE, when acting as a TLS client in the evaluated configuration, will only support TLSv1.2 protocols to connect and secure the following trusted channel:

- Connection from the TOE to the external Audit Server for audit data transfer

The TOE supports numerous ciphersuites, however, not all are used in the evaluated configuration. The ciphersuites not supported in the evaluated configuration must be disabled by the Security Administrator. The following ciphersuite is utilized in the evaluated configuration:

- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289

Mutual authentication support is not being claimed for the TOE when acting as a TLS client. When the TOE is placed in its evaluated configuration, the TSF supports Elliptic Curves and shall present the following Supported Elliptic Curves/Supported Groups Extensions in the Client Hello:

- secp384r1

The TOE will only establish a trusted channel if the peer certificate is valid. There is no administrative override mechanism to force the connection if the peer certificate is deemed invalid. The TOE, upon the presentation of the X.509v3 server host certificate, will validate the certificate per FIA_X509_EXT.1/REV requirements.

In the evaluated configuration, the TOE only supports Common Name (CN) and Subject Alternative Name (SAN) reference identifiers that are using IPv4 address values. Canonical formatting according to RFC 3986 is enforced. The TOE does not support the use of IPv6 addresses, URI, DNS (FQDN), service name reference identifiers, wildcards or pinned certificates.

The TSF converts that IP address, obtained from the certificate, from ASN.1 to the binary representation of the textual string of the IP address. The TSF also converts the IP address from the established network connection to the binary representation of the textual string of the IP address. The two representations are then compared to determine what action is performed next. The methodology for performing the check is as follows:

- If the SAN value exists:
  - If the two values match, revocation checking using the CRL is performed.
  - If the two values do not match, the certificate is deemed invalid and the connection is immediately terminated.
- If the SAN field is not used (non-existent), the representation of the CN value is used for comparison instead:
  - If the two values match, revocation checking using the CRL is performed.
  - If the two values do not match, the certificate is deemed invalid and the connection is immediately terminated.

The result of the certificate validation will be recorded in the audit log. There is no administrative override mechanism to force the connection if the peer certificate is deemed invalid.

8.2.13 FCS_TLSS_EXT.1

The TOE, when acting as a TLS server, will only support TLSv1.2 protocols to connect and secure the following trusted channels:

- Remote connection from the Remote Management Workstation to the TOE for administrative management

The TOE will deny connections from a client requesting any protocol versions besides TLS v1.2. When the TOE receives a TLS connection request with the wrong (unsupported) version, it returns a Fatal Alert: Handshake failure message and terminates the connection. Additionally, session resumption and session tickets are not supported.

The following ciphersuite is used for the evaluated configuration:

- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289

The TSF performs key establishment for TLS using ECDHE curve: secp384r1 and no other.

Mutual authentication is being claimed for the TOE, this discussion will be presented in Section 8.2.14. OpenSSL provides the cryptographic support for key establishment and encryption TLS channel when TOE acts as server.

8.2.14 FCS_TLSS_EXT.2

The TSF optionally supports TLS communications with mutual authentication for the Web GUI interface.

The TOE, upon the presentation of the X.509v3 client host certificate, will validate the certificate per FIA_X509_EXT.1/REV requirements when mutual authentication has been configured. The TSF shall verify that the presented identifier matches the reference identifier in the certificate.

In the evaluated configuration, the TOE only supports Common Name (CN) and Subject Alternative Name (SAN) reference identifiers that are using IPv4 address values. Canonical formatting according to RFC 3986 is enforced. The TOE does not support the use of IPv6 addresses, URI, DNS (FQDN), service name reference identifiers, wildcards or pinned certificates.

The TSF converts that IP address, obtained from the certificate, from ASN.1 to the binary representation of the textual string of the IP address. The TSF also converts the IP address from the established network connection to the binary representation of the textual string of the IP address. The two representations are then compared to determine what action is performed next. The methodology for performing the check is as follows:

- If the SAN value exists:
  - If the two values match, revocation checking using the CRL is performed.
  - If the two values do not match, the certificate is deemed invalid and the connection is immediately terminated.
- If the SAN field is not used (non-existent), the representation of the CN value is used for comparison instead:
  - If the two values match, revocation checking using the CRL is performed.
  - If the two values do not match, the certificate is deemed invalid and the connection is immediately terminated.

The result of the certificate validation will be recorded in the audit log. There is no fallback authentication fallback position if the certificate validation fails. There is no administrative override mechanism to force the connection if the peer certificate is deemed invalid.

8.3 Identification and Authentication

8.3.1 FIA_AFL.1

The TSF provides an administratively configurable counter threshold for consecutive failed password authentication attempts that will lock a user account for a defined period of time when the failure counter threshold is reached. The number consecutive failed password authentication attempts threshold can be configured through the Local CLI, Remote CLI, or Web GUI by a Security Administrator account. The failure threshold counter is configured on a per account basis.

The failure threshold counter must be configured to lock a user after 1-10 failed authentication attempts. The default setting of 0 must be changed upon initial configuration of the TOE for the default account of ADMIN and not be used when subsequent user accounts are created. The user account will automatically unlock after the configured time interval has passed. The Security Administrator can configure the lockout period between 0-99999 seconds. The default lockout time period is globally configured for all administrative interfaces to 86400 seconds (24 hours).

Additionally, an administrative account from any interface has the ability to unlock another administrative account in the event of an administrative account reaching the failed authentication attempts threshold.

A single failure counter is used, per user, across all interfaces (local, SSH, HTTPS). The failure counter increases with every failed login attempt, regardless of which interface is used, until the counter reaches its

administratively defined threshold. A successful password-based authentication occurring, through any interface, prior to the failure counter reaching its threshold will reset the failure counter to 0.

The TOE has a configurable option: "Serial Access Lockout:", where two options can be selected: "Allow Lock of All Users", and "Do Not Lock Admins". For the evaluated configuration the "Do Not Lock Admins" must be configured so the TOE does not lock the administrator role accounts on the serial physical interface (local access) but does lock the accounts from remote access.

8.3.2 FIA_PMG_EXT.1

In the evaluated configuration, the TOE's password security mode must be set to "enhanced". In the "enhanced" mode, the TOE supports the ability for a Security Administrator to set the minimum password length to 15 characters or greater with a maximum of 128 characters via any administrative interface. Passwords can be composed of any combination of upper and lower-case letters, numbers and special characters. The accepted special characters include: "!", "@", "#", "$", "%", "^", "(", ")", "_" , "+" , "|" , "~" , "{", " }" , "[" , "]" , "-", ".".

8.3.3 FIA_UAU.7

When authenticating to the TOE via the local CLI interface, the password feedback is obscured with no echo.

8.3.4 FIA_UAU_EXT.2 and FIA_UIA_EXT.1

The warning banner text can be configured by a Security Administrator. The display and acknowledgement of this banner is the only TOE functionality that is available to an unauthenticated user of the Web GUI, Remote CLI, and Local CLI.

When connecting remotely to the TOE's Web GUI via HTTPS, users must authenticate by providing their username/password credentials to the TOE. The TSF then verifies the credentials using a native authentication mechanism. The user must acknowledge the warning banner displayed before the authentication can proceed. The successful verification of the credentials presented to the TOE via HTTPS will provide the user access to all role-based functionality that is assigned to them for the Web GUI.

When connecting to the TOE remotely via an SSH connection to the Remote CLI, users can authenticate by providing their username/password credentials to the TOE. The TOE then verifies the credentials using a native authentication mechanism. Alternatively, the user can authenticate by providing a public-key for validation. The TSF will validate the public-key against the administratively imported and internally stored public-key assigned to that user requesting access. A successful verification of the credentials or public-key presented to the TOE via SSH will provide the user access to all role-based functionality that is assigned to them for the CLI.

When connecting to the TOE locally via a direct connection to the TOE platform, users must authenticate by providing their username/password credentials to the TOE. The TOE then verifies the credentials using a native authentication mechanism. The successful verification of the credentials presented to the TOE via a local connection will provide the user access to all role-based functionality that is assigned to them for the CLI.

8.3.5 FIA_X509_EXT.1/Rev, FIA_X509_EXT.2, and FIA_X509_EXT.3

The TOE uses X.509v3 certificates to support authentication for TLS connections to external IT entities in accordance with RFC 5280. The TOE performs certificate validity checking for any X.509v3 certificates presented to the TOE as part of TLS connections between itself and a remote audit server (audit log transmission) or HTTPS client (remote Web GUI administration) with mutual authentication enabled. The TOE also validates any X.509v3 certificate used to sign a software update during the software update process.

The TSF determines the validity of certificates by ensuring that the certificate path validation supports a minimum path length of three certificates and the certificate path is valid in accordance with RFC 5280. In addition:

- The TSF treats a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE
- The certificate path must terminate with a trusted CA certificate.
- The TSF validates a certificate path by ensuring the presence of the basicConstraints extension and that the CA flag is set to TRUE for all CA certificates.
- The TSF validates the extendedKeyUsage field according to the following rules:
    - Certificates used for trusted updates and executable code integrity verification must 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 must 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 must 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 must have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.
- The TSF rejects a certificate and not establish the connection if a certificate is found to be invalid for any reason prior to performing a revocation status check.
- The TSF performs revocation checks once the certificate is successfully validated per FIA_X509_EXT.1/REV.
- The TSF validates the certificate revocation status using a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3.
    - The CRL is downloaded at an administratively configurable frequency.
    - The CRL is immediately cached for use in determining the revocation status of the presented certificate.
    - The CRL is cached until the TOE successfully downloads a newer CRL from the CRL distribution point (CDP), replacing the currently cached value with the one it successfully retrieved.

- When the TSF cannot establish a connection to the CDP, the TOE will automatically continue attempting to download a new CRL at regular intervals until successful.

The TSF performs certificate revocation checking according to the following rules. These rules continue to apply when the TSF cannot establish a connection to download a new CRL:

- accept the certificate if the cached CRL is not yet expired and none of the certificates in the certificate chain (including the leaf certificate) are revoked.
- reject the certificate if the cached CRL is not yet expired and if the CRL identifies that any of the certificates in the certificate chain (including the leaf) are revoked. In this case, the TSF produces an audit record that reports an error message identifying the revoked certificate.
- reject the certificate if the cached CRL is expired regardless of the TOE's ability to successfully download a newer CRL from the CRL distribution point (CDP). In this case, the TSF produces an audit record that reports an error message identifying the certificate as invalid due to an expired CRL.
- The TSF does not provide a mechanism to override the validation decision.

An expired CRL does not automatically trigger a download of a new CRL. The CRL is updated according to the frequency defined by the administrator or via a manual update by the Security Administrator. The TSF follows the above rules for determining the revocation status of a certificate chain regardless of the TOE's ability to connect to the CDP.

A Certificate Request is generated as specified in RFC 2986 containing the public key and "Common Name" for the TOE to have its own certificate. The chain of certificates is validated up to a trusted root CA when the CA certificate response is received. For the TOE to authenticate to the external audit server, trusted CA certificates needed to validate the presented certificate(s) from the audit server must be installed into the TOE's certificate trust store.

8.4 Security Management

8.4.1 FMT_MOF.1/ManualUpdate, FMT_MTD.1/CoreData, FMT_MTD.1/Cryptokeys and FMT_SMF.1

The SFRs listed above have been combined to clarify the Security Management functions of the TOE including how the TOE implements authentication, identification, and also RBAC. The following description will also include restrictions for these roles and functions.

The TOE utilizes role-based access control (RBAC), as described in FMT_SMR.2, to restrict access to the administrative functions that manage the TSF data. Display and acknowledgement of a warning banner is the only TOE functionality available prior to identification and authentication. The TOE limits the presented functionality based on the privileges bound to the authenticated user. The available functionality presented to an authenticated user is based on the group of permissions and the privileges associated with the permissions aligned to the authenticated user's assigned role. These permissions/privileges are bound to the user only after the user has successfully authenticated. The TSF restricts the ability to manage the TSF data to only Security Administrators.

The role of Security Administrator is fulfilled by both the "Administrator" user role and for trusted updates only the "Provision" user role.

The TSF management functions that are restricted to Security Administrators based on local or remote administration, and scoped by this evaluation are:

Management Function	Local CLI (Physical Connection)	Remote CLI (SSH)	Web GUI (HTTPS)
Configure Banner Text	Administrator	Administrator	Administrator
Configure Idle Session Timeout	Administrator	Administrator	Administrator
Initiate Manual Update	Administrator Provision	Administrator Provision	Administrator Provision
Configure Failed Lockout Threshold	Administrator	Administrator	Administrator
Configure Lockout Duration	Administrator	Administrator	Administrator
Manage the cryptographic keys	Administrator	Administrator	Administrator
Configure the cryptographic functionality	Administrator	Administrator	Administrator
Re-enable Administrator accounts	Administrator	Administrator	Administrator
Set time	Administrator	Administrator	Administrator
Configure NTP	Administrator	Administrator	Administrator
Manage the TOE's trust store and designate X.509v3 certificates as trust anchors (i.e., generate, import, delete of X.509 certificates)	Administrator	Administrator	Administrator
Ability to manage the trusted public keys database (i.e., import and deletion of SSH keys public keys)	Administrator	Administrator	Administrator

Table 20: Management Functions to Management Interface Identification

8.4.2 FMT_SMR.2

The TOE support numerous types of user roles. The TOE is designed to use permissions which allow, limit or prevent user access to specific administrative tools based on the aligned user role. Upon successful authentication, the TSF associates the administratively defined set of permissions (role) for that user to the subject acting on behalf of that user. The TSF enforces role-based access control (RBAC) to limit access to TSF functions and data based on the set of permissions bound to the subject.

The TOE has two administrative roles for the PP defined management functions:

- Administrator – has the ability to perform all PP defined management functions
- Provision – administrative abilities are limited to updating TOE software

8.5 Protection of the TSF

8.5.1 FPT_APW_EXT.1

No authentication passwords are stored by the TOE in plaintext. All authentication passwords are hashed using SHA-512. There is no function provided by the TOE to display a password value in plaintext nor is the password data recoverable.

8.5.2 FPT_SKP_EXT.1

The TSF prevents unauthorized disclosure of pre-shared keys, symmetric keys and private keys as it does not provide any interface mechanism (CLI or Web GUI) to view these items from volatile memory or file system storage. However, Security Administrators that have the ability to escalate to root privileges, using the sudo command, can have authorized access to the file locations where the secret keys, private keys, and secret key data are stored.

8.5.3 FPT_STM_EXT.1

The TOE provides its own time via its internal clock that can be adjusted manually by a Security Administrator via the Web GUI. The TOE can also be configured to use an NTP Server as a time source. See Section 8.2.8 for details regarding the TOE's utilization of NTP as a time source.

The TOE uses the clock for several security-relevant purposes, including:

- Audit record timestamps (seconds, milliseconds, microseconds, or nanoseconds).
- X.509v3 certificate validation
- Inactivity of remote sessions
- Inactivity of local session

8.5.4 FPT_TST_EXT.1

The TOE performs the following self-check procedures before starting the operating system to assure integrity of the filesystem, TOE software, and cryptographic functions.

Standard Linux Filesystem check:

The TOE performs the following checks of the file system:

- mounts (creates) basic virtual RAM file systems
- verifies and mounts the non-volatile file system
- verifies and mounts the active or standby software partition file system

Failures for any of these checks may result in entering a non-operational state. The TOE is designed to automatically attempt to fix and continue if errors are found.

Software Integrity Check:

The TOE validates software integrity on the filesystem by verifying the current state of the constant files on the root partition against the manifest file that was generated and included in the software as part of the build process. The manifest contains the following information:

- executable binary files
- executable text files (scripts);
- shared libraries
- all constant files on root partition
- SHA-384 hashes for comparison
- Ownership for comparison
- file permissions for comparison

This check will result in errors that indicate an integrity issue with one or more of the TOE's software files. A failure of this check results in the non-operational state.

Cryptographic Check:

The TOE executes Known Answer Tests for the following cryptographic functionalities:

- CTR_DRBG
- AES256-GCM
- Diffie-Hellman Safe Primes Key Generation
- Diffie-Hellman Safe Primes Key Verification
- Diffie-Hellman Safe Primes Shared Secret Computation
- ECDSA Key Generation
- ECDSA Key Verification
- ECDSA Signature Verification
- HMAC-SHA-384
- SHA-384
- SHA-512

Based on FIPS 140-2 methodology, failure of the cryptographic checks will result in the TOE NOT performing any cryptographic services. This check results in errors identifying the failed cryptographic operations. A failure of this check results in the non-operational state.

Analysis of Self-Tests:

These tests are sufficient to validate the correct operation of the TOE because the self-tests are designed to discover any anomalies that would cause the software to be executed in an unpredictable or inconsistent manner. These tests provide assurance that the software has not been tampered with, the filesystem is mounted and validated, and the cryptography is operating correctly.

POST Execution Report:

The TOE generates a POST Execution Report which is available to administrators in a dedicated file accessible from a shell, from CLI, or Web GUI. This report will also be sent to the syslog server when the TOE completes its startup. The report contains the following information for each test executed:

- test name or identifier
- test result (success or failure)
- date and time

Non-Operational State:

In the non-operational state, the front NCU panel blinks with alternating yellow-red. The system does not provide access via any remote administration interface. However, access to the system can be achieved through a serial RJ45 interface. Upon access, the TOE:

- Provides a POST execution report
- Allows access to the system shell
- Allows you to reboot the system
- Allows you to try to reach the operational state

8.5.5 FPT_TUD_EXT.1 and FPT_TUD_EXT.2

The currently executing version of the TOE's software is displayed immediately following successful authentication on all administrative interfaces.

The Web GUI displays the currently executing version of the TOE's software in the header of the Web GUI page in the top left corner of the screen. A Security Administrator can also view the currently executing version of the TOE's software by navigating to the Node's page. Once in the Node's page, a Security Administrator must navigate to the "Software" page in the options tree on the left hand side of the page and click "NCU". The "Active Software Release" tab in the NCU page will display the current executing version of the TOE's software.

For both the Remote CLI (SSH) and the Local CLI, the currently executing version of the TOE's software is displayed in a banner along the top of the options menu screen. This options menu is shown following a Security Administrator's successful authentication to the TOE. A Security Administrator can also view the currently executing version of the TOE's software by navigating through the "System Management" page followed by the "Software & Database Control" page. The software version is available under the "NCU Updates" tab. Additionally, the standby image of the TOE can be viewed on this page under the heading "STBY".

The TOE does not automatically check for software and firmware updates for the system. The Security Administrator must download the TOE's update image from the Adtran Customer Portal page to the application server or local workstation. The administrator must use a computer separate from the TOE to recompute the hash of the downloaded image and verify it matches the published hash obtained from the Customer Portal page. Once this validation is complete, the administrator must sign the validated software, using the end user's approved code signing X.509v3 certificate. This creates the trusted update package. The trusted updated package is then placed on the customer's file server. The administrator must import the certificate authority (CA) certificates for the code signing certificate and mark the certificate as trusted.

The administrator must fetch the trusted update package from the application server or local workstation using the Web GUI. Upon downloading, the TSF will validate the package. If the package validation is successful the trusted update is loaded into the standby area where it will reside dormant until the administrator activates that image (delayed activation). If the validation fails the package is deleted from the TOE.

The TSF validates the package by validating the code signing certificate inside the package using the rules outline in FIA_X509_EXT.1/REV, including the CRL revocation checking, and then verifying the digital signature that was applied to software package. The determination to place the code into the standby area is based on the following:

- If the certificate is deemed invalid (e.g. expired or revoked), the image is not installed and is removed from the system.
- If the certificate is deemed valid, the TSF will then validate the digital signature applied to the code:
    - If the digital signature is not valid, the image is not installed and is removed from the system.
    - If the digital signature check succeeds, the software image is placed in the Standby Area.

The administrator initiates the activation of the image by navigating to the "Nodes" page and then the "Software" page. Once in the Software page, the administrator will locate the "Activate" option in the "Activate Software in Standby Area".

The currently executing version of the TOE is displayed as well as the version of the image in the standby area. The previous version of the TOE's software is still available for reactivation on the system via the security administrator at any time. Two images remain on the machine until the standby image is either deleted or replaced.

8.6 TOE Access

8.6.1 FTA_SSL_EXT.1

The TOE will automatically terminate a local session on the Local CLI interface due to inactivity according to a session inactivity timer value set by the TOE's Security Administrator. The Security Administrator can configure the local session inactivity timer via the CLI. The inactivity time period for a local session can be configured between 30 – 3600 seconds.

A successful automatic session termination of a local user session can be verified through the appearance of a login prompt/notification banner. Once a session has automatically terminated, the user will be required to reauthenticate to the TOE and open a new user session.

8.6.2 FTA_SSL.3

The TOE will terminate a remote session for both the Remote CLI and Web GUI interfaces due to inactivity according to each interface's respective session inactivity timer configuration. The inactivity timers are configured by the TOE's Security Administrator via the CLI. The inactivity time period for a remote session can be configured between 30 – 3600 seconds.

A successful automatic session termination of a remote user sessions for both the CLI and Web GUI interfaces can be verified through the appearance of a login prompt/notification banner. Once a session has automatically terminated, the user will be required to reauthenticate to the TOE and open a new user session.

8.6.3 FTA_SSL.4

Any user accessing the TOE is capable of terminating their own session. A Web GUI user may terminate their own sessions by pressing "Logout" under the account button in the top right corner of the screen. A Local and Remote CLI user may terminate their own session by navigating the menu and selecting "Quit".

For all administrative interfaces, a manual user session termination can be verified through the appearance of a login prompt. Once a session has automatically terminated, the user will be required to reauthenticate to the TOE and open a new user session.

8.6.4 FTA_TAB.1

There are three possible administrative ways to log into the TOE: locally via physical connection to access the Local CLI, remotely via SSH connection to access the Remote CLI, and remotely using the Web GUI which establishes a HTTPS connection. When logging in locally or remotely, the pre-authentication banner is displayed and must be acknowledged prior to authentication. The authentication banner for the administrative interfaces is customizable by the Security Administrator.

8.7 Trusted Path/Channels

8.7.1 FTP_ITC.1

The TOE provides the ability to secure sensitive data in transit to and from the Operational Environment. The TOE, acting as the TLS client, uses the TLS protocol to initiate and establish the trusted channel to support the following capabilities:

- Export generated TOE audit data to an external Audit Server

The TOE does not support mutual authentication between itself and the Audit Server. The TOE's TLS client implementation is conformant to FCS_TLSC_EXT.1. TLS communications use X.509v3 certificates to support authentication.

8.7.2 FTP_TRP.1/Admin

Remote administration of the TOE is secured by the utilization of SSH and HTTPS protocols.

An HTTPS connection is used for establishing a connection from the Remote Management Workstation to the TOE's Web GUI. The HTTPS connection enables remote administration of the TOE via the Remote Management Workstation. The TOE acts as the HTTPS server and is conformant to the requirements stated in FCS_HTTPS_EXT.1

An SSH connection is used for establishing a connection from the Remote Management Workstation to the TOE's Remote CLI. The SSH connection enables remote administration of the TOE via the Remote Management Workstation. The TOE acts as the SSH server and is conformant to the requirements stated in FCS_SSHS_EXT.1.