ciena : the network specialist

Ciena 5400 Series Packet Optical Platform

Security Target

ST Version: 1.0
January 11, 2016

Ciena Corporation
7035 Ridge Road
Hanover, MD 21076

Prepared By:

Booz | Allen | Hamilton
delivering results that endure

Cyber Assurance Testing Laboratory
900 Elkridge Landing Road, Suite 100
Linthicum, MD 21090

Table of Contents

Table of Figures

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.

1.1.1 ST Identification

ST Title:	Ciena 5400 Series Packet Optical Platform Security Target
ST Version:	1.0
ST Publication Date:	January 11, 2016
ST Author:	Booz Allen Hamilton

1.1.2 Document Organization

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

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

Chapter 3 describes the conformance claims made by this ST.

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

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

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

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

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

1.1.3 Terminology

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

Term	Definition
Account Administrator	Account Administrator, or AA, is the highest administrative privilege available on the TOE's TL1 interface. All TSF-relevant functionality that can be managed from the TL1 interface can be performed by the AA. This role is an Authorized Administrator for the TL1 interface.
MCLI	Management Command Line Interface (MCLI) is a command shell interface that can be used to administer the TOE locally or remotely using SSH. This interface is primarily used for functions that are performed during initial setup/deployment of the TOE.
Superuser	Superuser is the only administrative privilege available on the TOE's MCLI. All TSF-relevant functionality that can be managed from the MCLI can be performed by the superuser. This role is an Authorized Administrator for the MCLI.
TL1 [Management Interface]	The Transaction Language 1 (TL1) management interface is a TL1-compatible command shell interface that can be used to administer the TOE locally or remotely using SSH. This interface is distinct from the MCLI and is used to perform functions that may be modified during ongoing administration of the TOE.

Table 1-1: Customer Specific Terminology

Term	Definition
Authorized Administrator	The claimed Protection Profile defines an Authorized Administrator role that is authorized to manage the TOE and its data.
Entropy	A string of quasi-random data that is generated by unpredictable physical and/or logical phenomena in a computer and is used in the generation of random numbers.
Security Administrator	Synonymous with Authorized Administrator.
Trusted Channel	An encrypted connection between the TOE and a trusted remote server.
Trusted Path	An encrypted connection between a remote administrative interface and the TOE.

Table 1-2: CC Specific Terminology

1.1.4 Acronyms

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

Acronym	Definition
AES	Advanced Encryption Standard
CAVP	Cryptographic Algorithm Validation Program
CBC	Cipher Block Chaining
CLI	Command Line Interface
CSP	Critical Security Parameter
DHE	Diffie-Hellman
DRBG	Deterministic Random Bit Generator
HMAC	Hashed Message Authentication Code

MCLI	Management Command Line Interface
MPLS	Multiprotocol Label Switching
NDPP	Network Device Protection Profile
NTP	Network Time Protocol
OSI	Open Systems Interconnection
OTN	Optical Transport Network
RSA	Rivest Shamir Adelman (encryption algorithm)
SDH	Synchronous Digital Hierarchy
SFTP	Secure File Transfer Protocol
SHA	Secure Hash Algorithm
SHS	Secure Hash Standard
SONET	Synchronous Optical Networking
SSH	Secure Shell
TL1	Transaction Language One

Table 1-3: Acronym Definition

1.2 TOE Reference

The TOE is the Ciena 5400 Series Packet Optical Platform, which is a packet-optical switching platform. It is also known as the Ciena 5400 Series. The 5400 Series contains two models: the Ciena 5430 and Ciena 5410. Each of these devices runs Linux kernel version 3.4.36 and provides identical security functionality to one another.

1.3 TOE Overview

The Ciena 5400 Series Packet Optical Platform is a family of hardware devices that provides OSI Layer 2 network traffic management services. It is a packet-optical switching platform that enables users to direct traffic to designated ports, giving them control of network availability for specific services. The system features an agnostic switch fabric that is capable of switching SONET/SDH, OTN, and Ethernet/MPLS networks.

The Ciena 5400 Series supports OC-48/STM-16 and OC-192/STM-64, OTU1/2/3/4, and 1/10/40/100G Ethernet interfaces to provide up to 15 Tbps switching capacity using a combination of:

- Up to 30 line modules on the 5430 chassis
- Up to 10 line modules on the 5410 chassis

The Ciena 5400 Series is a family of standalone single hardware appliances that run Linux. The Target of Evaluation (TOE) is the general network device security functions that are provided by the Ciena 5400 Series, such as security auditing, trusted communications, security management, and identification and authentication. The appliances provide command line and TL1 interfaces to the TOE’s security functionality as well as the switching behavior that is beyond the scope of the claimed Protection Profile.

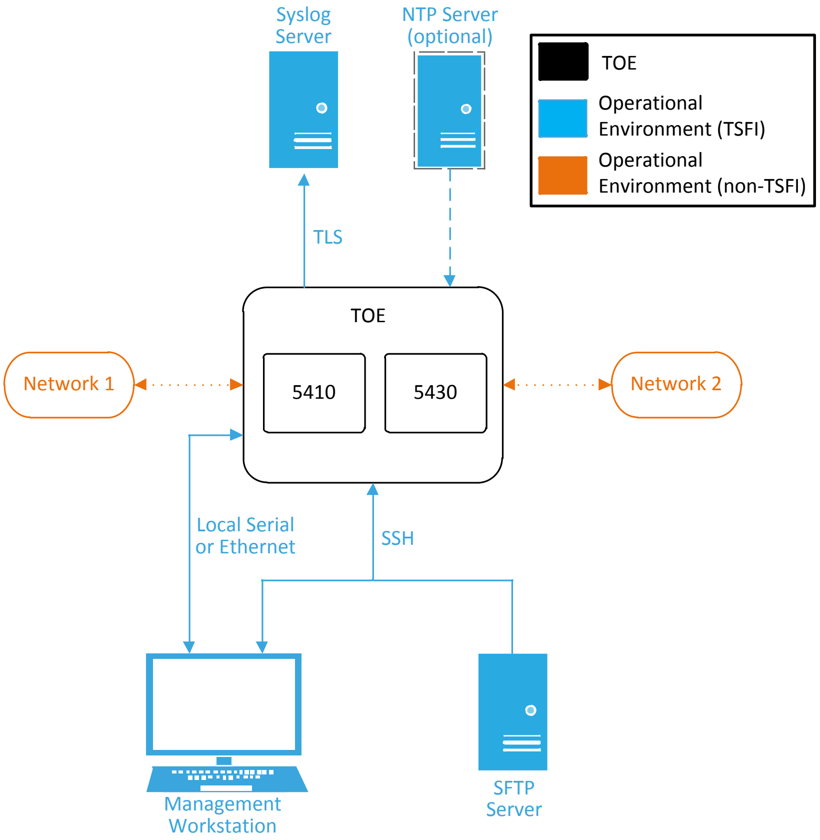

Figure 1-1: TOE Boundary

In practice, the TOE will be deployed to perform OSI Layer 2 switching functions and will be connected to a number of other network traffic infrastructure equipment. This has not been depicted in detail because this capability is out of scope of the TOE from a security functional perspective.

1.4 TOE Type

The TOE type for the Ciena 5400 Series is Network Device. The TOE is a hardware appliance whose primary functionality is related to the handling of network traffic. The NDPP defines a network device as "a device composed of hardware and software that is connected to the network and has an infrastructure role in the overall enterprise." Additionally, the NDPP says that example devices that fit this definition include routers, firewalls, intrusion detection systems, audit servers, and switches that have Layer 3 functionality. The TOE is a switch that has Layer 2 and Layer 3 functionality. The TOE type is justified because the TOE provides an infrastructure role in internetworking of different network environments across an enterprise.

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 TOE is the Ciena 5400 Series Packet Optical Platform. This is a family of products that contains the following hardware models:

- Ciena 5410 Packet Optical Platform
- Ciena 5430 Packet Optical Platform

Each of these hardware models is a standalone network appliance.

2.2 Components and Applications in the Operational Environment

The following table lists components and applications in the environment that the TOE relies upon in order to function properly:

Component	Definition
Management Workstation	Any general-purpose computer that is used by an administrator to manage the TOE. The TOE can be managed remotely, in which case the management workstation requires an SSH client, or locally, in which case the management workstation must be physically connected to the TOE using the serial port and must use a terminal emulator that is compatible with serial communications.
NTP Server	A system that provides an authoritative and reliable source of time using network time protocol (NTP).
Syslog Server	A general-purpose computer that is running a syslog server, which is used to store audit data generated by the TOE.
Update Server	A server running the secure file transfer protocol (SFTP) that is used as a location for storing product updates that can be transferred to the TOE.

Table 2-1: Evaluated Components of the Operational Environment

In the evaluated configuration, an NTP server is optional, as the TOE also provides the ability to maintain system time using its internal hardware clock.

2.3 Excluded from the TOE

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

2.3.1 Not Installed

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

2.3.2 Installed but Requires a Separate License

No components are installed that require a separate license.

2.3.3 Installed But Not Part of the TSF

This section contains functionality or components that are part of the purchased product but are not part of the TSF relevant functionality that is being evaluated as the TOE.

- CORBA administrative interface – by default, the CORBA administrative interface that can be used to interact with the TSF does not provide security. In the evaluated configuration, it will be disabled following initial setup so that all remote administrative communications use SSH.
- FTP, HTTP, TELNET, TELNET_TLS, SNMP – these protocols must be locked (disabled) in the evaluated configuration.

Additionally, the TOE includes a number of functions 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 physical boundary of the TOE includes the Ciena 5410 and 5430 Packet Optical Platform hardware appliances and the software that runs on them. The TOE includes a Freescale MPC8572 processor which is used to provide entropy to the software deterministic random bit generation function.

The TOE guidance documentation that is considered to be part of the TOE can be found in the Common Criteria-specific guidance for the Ciena 5400 Series Packet Optical Platform, which is delivered on physical media to customers purchasing the equipment and is also made available on the Ciena website.

2.4.1 Software

The operating system used by the TOE is Linux, kernel version 3.4.36. The TOE is managed using a combination of a management command-line interface (MCLI) and Transaction Language 1 (TL1) interface. Both of these interfaces can be used for either local administration or secure remote administration using SSH.

2.5 Logical Boundary

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

1. Security Audit
2. Cryptographic Support
3. User Data Protection
4. Identification and Authentication
5. Security Management
6. Protection of the TSF
7. TOE Access
8. Trusted Path/Channels

2.5.1 Security Audit

The TOE provides extensive auditing capabilities. The security log includes detailed records of all user activity including events related to authentication, management, and session termination. Establishment, termination, and failure to establish trusted communications is also audited. The TOE generates audit logs using syslog, and the collected audit data can be transmitted securely to a remote server in the Operational Environment.

The TOE records, for each audited event, the date and time of the event, the type of event, the subject's claimed identity, and the outcome (success or failure) of that event. Depending on the specific type of event, additional data may be included in the audit record.

2.5.2 Cryptographic Support

The TOE provides cryptography in support of SSH and TLS trusted communications for remote administration, remote storage of audit data, and secure download of TOE updates. Asymmetric keys used by the TSF are generated in accordance with NIST SP 800-56. The TOE uses CAVP-validated cryptographic algorithms (certificates AES #3753, RSA #1930, SHS #3124, HMAC #2456, DRBG #1029) to ensure that appropriately strong cryptographic algorithms are used for these trusted communications.

The TOE collects entropy from a third-party hardware source contained within the device to ensure sufficient randomness for secure key generation.

2.5.3 User Data Protection

The TOE ensures that packets transmitted from the TOE do not contain residual information from previous packets. Any data that terminates before the minimum packet size is reached is padded with zeroes.

2.5.4 Identification and Authentication

All users must be identified and authenticated to the TOE via locally-defined username and password or username and SSH public key before being allowed to perform any actions on the TOE, except viewing a banner. The TOE provides complexity rules that ensure that user-defined passwords will meet a minimum security strength through the set of supported characters and configurable minimum password length. As part of connecting to the TOE locally, using the management workstation, password data will be obfuscated as it is being input.

2.5.5 Security Management

The product maintains several pre-defined roles for the TL1 administrative interface. Of these, the Account Administrator (AA) is the only administrative role that has the ability to manage the TSF, so it is the only TL1 role that is within the scope of the TOE. The TOE also provides a separate superuser role that is used exclusively for managing the TSF using the MCLI. The superuser and AA roles are analogous to the role of Security Administrator as defined by the NDPP. The remaining roles perform network management related functionality that is not considered to be part of the TSF.

2.5.6 Protection of the TSF

The TOE is expected to ensure the security and integrity of all data that is stored locally and accessed remotely. The TOE stores passwords in an obfuscated format. The cryptographic module prevents the unauthorized disclosure of secret cryptographic data, and administrative passwords are hashed using SHA-256. The TOE maintains system time with either its local hardware clock or optionally with an NTP server synchronization. TOE software updates are acquired using SFTP and initiated using the MCLI. Software updates are digitally signed to ensure their integrity. The TSF also validates its correctness through the use of self-tests for both cryptographic functionality and integrity of the system software.

2.5.7 TOE Access

The TOE can terminate inactive sessions after an administrator-configurable time period. The TOE also allows users to terminate their own interactive session. Once a session has been terminated the TOE requires the user to re-authenticate to establish a new session.

The TOE can also display a configurable banner on both the MCLI and TL1 interfaces that is displayed prior to use of any other TSF.

2.5.8 Trusted Path/Channels

The TOE establishes a trusted path to the TOE using SSH for MCLI and TL1 administration. The TOE also establishes trusted channels for sending audit data to a remote syslog server using TLS and for downloading software updates and manually transferring audit records using SFTP (FTP over SSH).

3 Conformance Claims

3.1 CC Version

This ST is compliant with Common Criteria for Information Technology Security Evaluation, Version 3.1 Revision 4 September 2012.

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 11 January 2016.

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 11 January 2016.

3.4 PP Claims

This ST claims exact conformance to the following Protection Profile:

- Protection Profile for Network Devices, version 1.1 [NDPP]

This PP claim also includes the NDPP Errata #3 that provides updates and clarifications to the NDPP.

3.5 Package Claims

The TOE claims exact conformance to the NDPP, version 1.1.

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

- FCS_SSH_EXT.1
- FCS_TLS_EXT.1

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

3.6 Package Name Conformant or Package Name Augmented

This ST and TOE are conformant with the claimed PP.

3.7 Conformance Claim Rationale

The NDPP states the following: “This is a Protection Profile (PP) for a network device. A network device in the context of this PP is a device composed of hardware and software that is connected to the network and has an infrastructure role in the overall enterprise. Examples of a ‘network device’ that should claim compliance to this PP include routers, firewalls, IDSs, audit servers, and switches that have Layer 3 functionality.”

The TOE is a family of hardware appliances that is designed to perform low-level network traffic switching between SONET/SDH, OTN, and Ethernet/MPLS switches. As such, it can be understood as a network switch. Therefore, the conformance claim is appropriate.

4 Security Problem Definition

4.1 Threats

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

Threat	Threat Definition
T.ADMIN_ERROR	An administrator may unintentionally install or configure the TOE incorrectly, resulting in ineffective security mechanisms.
T.TSF_FAILURE	Security mechanisms of the TOE may fail, leading to a compromise of the TSF.
T.UNDETECTED_ACTIONS	Malicious remote users or external IT entities may take actions that adversely affect the security of the TOE. These actions may remain undetected and thus their effects cannot be effectively mitigated.
T.UNAUTHORIZED_ACCESS	A user may gain unauthorized access to the TOE data and TOE executable code. A malicious user, process, or external IT entity may masquerade as an authorized entity in order to gain unauthorized access to data or TOE resources. A malicious user, process, or external IT entity may misrepresent itself as the TOE to obtain identification and authentication data.
T.UNAUTHORIZED_UPDATE	A malicious party attempts to supply the end user with an update to the product that may compromise the security features of the TOE.
T.USER_DATA_REUSE	User data may be inadvertently sent to a destination not intended by the original sender.

Table 4-1: TOE Threats

4.2 Organizational Security Policies

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

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

Table 4-2: 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 NDPP.

Assumption	Assumption Definition
A.NO_GENERAL_PURPOSE	It is assumed that 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.
A.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the environment.
A.TRUSTED_ADMIN	TOE Administrators are trusted to follow and apply all administrator guidance in a trusted manner.

Table 4-3: TOE Assumptions

4.4 Security Objectives

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

4.4.1 TOE Security Objectives

This section identifies the security objectives of the TOE. These objectives have been taken directly from the NDPP.

Objective	Objective Definition
O.PROTECTED_COMMUNICATIONS	The TOE will provide protected communication channels for administrators, other parts of a distributed TOE, and authorized IT entities.
O.VERIFIABLE_UPDATES	The TOE will provide the capability to help ensure that any updates to the TOE can be verified by the administrator to be unaltered and (optionally) from a trusted source.
O.SYSTEM_MONITORING	The TOE will provide the capability to generate audit data and send those data to an external IT entity.
O.DISPLAY_BANNER	The TOE will display an advisory warning regarding use of the TOE.
O.TOE_ADMINISTRATION	The TOE will provide mechanisms to ensure that only administrators are able to log in and configure the TOE, and provide protections for logged-in administrators.
O.RESIDUAL_INFORMATION_CLEARING	The TOE will ensure that any data contained in a protected resource is not available when the resource is reallocated.
O.SESSION_LOCK	The TOE shall provide mechanisms that mitigate the risk of unattended sessions being hijacked.
O.TSF_SELF_TEST	The TOE will provide the capability to test some subset of its security functionality to ensure it is operating properly.

Table 4-4: TOE Objectives

4.4.2 Security Objectives for the Operational Environment

The TOE’s operating environment must satisfy the following objectives:

Objective	Objective Definition
OE.NO_GENERAL_PURPOSE	There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE.
OE.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.
OE.TRUSTED_ADMIN	TOE Administrators are trusted to follow and apply all administrator guidance in a trusted manner.

Table 4-5: 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. Therefore the “Extended” used in SFR component name will be dropped.

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 and italicized 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.

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	External Audit Trail Storage
Cryptographic Support	FCS_CKM.1	Cryptographic Key Generation (for asymmetric keys)
	FCS_CKM_EXT.4	Cryptographic Key Zeroization
	FCS_COP.1(1)	Cryptographic Operation (for data encryption/decryption)
	FCS_COP.1(2)	Cryptographic Operation (for cryptographic signature)
	FCS_COP.1(3)	Cryptographic Operation (for cryptographic hashing)
	FCS_COP.1(4)	Cryptographic Operation (for keyed-hash message authentication)
	FCS_RBG_EXT.1	Cryptographic Operation (Random Bit Generation)
	FCS_SSH_EXT.1	SSH
	FCS_TLS_EXT.1	TLS
User Data Protection	FDP_RIP.2	Full Residual Information Protection
Identification and	FIA_PMG_EXT.1	Password Management

Class Name	Component Identification	Component Name
Authentication	FIA_UAU_EXT.2	Password-based Authentication Mechanism
	FIA_UAU.7	Protected Authentication Feedback
	FIA_UIA_EXT.1	User Identification and Authentication
Security Management	FMT_MTD.1	Management of TSF Data (for general 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.1	Reliable Time Stamps
	FPT_TST_EXT.1	TSF Testing
	FPT_TUD_EXT.1	Trusted Update
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 Banners
Trusted Path /Channels	FTP_ITC.1	Inter-TSF Trusted Channel
	FTP_TRP.1	Trusted Path

Table 6-1: Security Functional Requirements for the TOE

6.3 Security Functional Requirements

6.3.1 Class FAU: Security Audit

6.3.1.1 *FAU_GEN.1 Audit Data Generation*

FAU_GEN.1.1 The TSF shall be able to generate an audit record of the following auditable events:

a) Start-up and shut-down of the audit functions;
b) All auditable events for the not specified level of audit; and
c) All administrative actions;
d) [Specifically defined auditable events listed in ***Table 6-2***].

FAU_GEN.1.2 The TSF shall record within each audit record at least the following information:

a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event; and
b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [information specified in column three of ***Table 6-2***].

Requirements	Auditable Events	Additional Audit Record Contents
FCS_SSH_EXT.1	Failure to establish an SSH session Establishment/Termination of an SSH session.	Reason for failure. Non-TOE endpoint of connection (IP address) for both successes and failures.
FCS_TLS_EXT.1	Failure to establish an TLS session Establishment/Termination of an TLS session.	Reason for failure. Non-TOE endpoint of connection (IP address) for both successes and failures.
FIA_UIA_EXT.1	All use of the identification and authentication mechanism.	Provided user identity, origin of the attempt (e.g., IP address).
FIA_UAU_EXT.2	All use of the authentication mechanism.	Origin of the attempt (e.g., IP address).
FPT_STM.1	Changes to the time.	The old and new values for the time. Origin of the attempt (e.g., IP address).
FPT_TUD_EXT.1	Initiation of update.	No additional information.
FTA_SSL_EXT.1	Any attempts at unlocking of an interactive session.	No additional information.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	No additional information.
FTA_SSL.4	The termination of an interactive session.	No additional information.
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	Initiation of the trusted channel. Termination of the trusted channel. Failures of the trusted path functions.	Identification of the claimed user identity.

Table 6-2: Auditable Events

Application Note: *The TSF only terminates interactive sessions and does not lock them (see FTA_SSL_EXT.1). Therefore, the auditable event of 'any attempts at unlocking of an interactive session' is synonymous with authentication attempts to the TOE.*

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 External Audit Trail 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 implementing the [SSH, TLS] protocol.

6.3.2 Class FCS: Cryptographic Support

6.3.2.1 *FCS_CKM.1 Cryptographic Key Generation (for asymmetric keys)*

FCS_CKM.1.1 The TSF shall generate asymmetric cryptographic keys used for key establishment in accordance with

[NIST Special Publication 800-56A, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" for finite field-based key establishment schemes;

NIST Special Publication 800-56B, "Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography" for RSA-based key establishment schemes]

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

6.3.2.2 *FCS_CKM_EXT.4 Cryptographic Key Zeroization*

FCS_CKM_EXT.4.1 The TSF shall zeroize all plaintext secret and private cryptographic keys and CSPs when no longer required.

6.3.2.3 *FCS_COP.1(1) Cryptographic Operation (for data encryption/decryption)*

FCS_COP.1.1(1) The TSF shall perform encryption and decryption in accordance with a specified cryptographic algorithm AES operating in [CBC, [*no other modes*]] and cryptographic key sizes 128-bits and 256-bits that meets the following:

- FIPS PUB 197, "Advanced Encryption Standard (AES)"
- [NIST SP 800-38A].

6.3.2.4 *FCS_COP.1(2) Cryptographic Operation (for cryptographic signature)*

FCS_COP.1.1(2) The TSF shall perform cryptographic signature services in accordance with a [(2) RSA Digital Signature Algorithm (rDSA) with a key size (modulus) of 2048 bits or greater**]**

that meets the following:

Case: RSA Digital Signature Algorithm

- FIPS PUB 186-2 or FIPS PUB 186-3, “Digital Signature Standard”

6.3.2.5 *FCS_COP.1(3) Cryptographic Operation (for cryptographic hashing)*

FCS_COP.1.1(3) The TSF shall perform cryptographic hashing services in accordance with a specified cryptographic algorithm [SHA-1, SHA-256] and message digest sizes [160, 256] bits that meet the following: FIPS PUB 180-3, “Secure Hash Standard.”

6.3.2.6 *FCS_COP.1(4) Cryptographic Operation (for keyed-hash message authentication)*

FCS_COP.1.1(4) The TSF shall perform keyed-hash message authentication in accordance with a specified cryptographic algorithm HMAC-[SHA-1, SHA-256], key size [*greater than block size, less than block size, equal to block size*], and message digest sizes [160, 256] bits that meet the following: FIPS PUB 198-1, “The Keyed-Hash Message Authentication Code”, and FIPS PUB 180-3, “Secure Hash Standard.”

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

FCS_RBG_EXT.1.1 The TSF shall perform all random bit generation (RBG) services in accordance with [NIST Special Publication 800-90 using [Hash_DRBG (any)]]; seeded by an entropy source that accumulated entropy from [a TSF-hardware-based noise source].

FCS_RBG_EXT.1.2 The deterministic RBG shall be seeded with a minimum of [256 bits] of entropy at least equal to the greatest security strength of the keys and hashes that it will generate.

6.3.2.8 *FCS_SSH_EXT.1 SSH*

FCS_SSH_EXT.1.1 The TSF shall implement the SSH protocol that complies with RFCs 4251, 4252, 4253, 4254, and [no other RFCs].

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

FCS_SSH_EXT.1.3 The TSF shall ensure that, as described in RFC 4253, packets greater than [*32000*] bytes in an SSH transport connection are dropped.

FCS_SSH_EXT.1.4 The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms: AES-CBC-128, AES-CBC-256, [no other algorithms].

FCS_SSH_EXT.1.5 The TSF shall ensure that the SSH transport implementation uses [SSH_RSA] and [no other public key algorithms] as its public key algorithm(s).

FCS_SSH_EXT.1.6 The TSF shall ensure that data integrity algorithms used in SSH transport connection is [hmac-sha1, hmac-sha2-256].

FCS_SSH_EXT.1.7 The TSF shall ensure that diffie-hellman-group14-sha1 and [no other methods] are the only allowed key exchange methods used for the SSH protocol.

6.3.2.9 *FCS_TLS_EXT.1 TLS*

FCS_TLS_EXT.1.1 The TSF shall implement one or more of the following protocols [TLS 1.2 (RFC 5246)] supporting the following ciphersuites:

Mandatory Ciphersuites:

TLS_RSA_WITH_AES_128_CBC_SHA

Optional Ciphersuites:

[TLS_RSA_WITH_AES_256_CBC_SHA

TLS_RSA_WITH_AES_128_CBC_SHA256

TLS_RSA_WITH_AES_256_CBC_SHA256

].

6.3.3 Class FDP: User Data Protection

6.3.3.1 *FDP_RIP.2 Full Residual Information Protection*

FDP_RIP.2.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the [allocation of the resource to, deallocation of the resource from] all objects.

6.3.4 Class FIA: Identification and Authentication

6.3.4.1 *FIA_PMG_EXT.1 Password Management*

FIA_PMG_EXT.1.1 The TSF shall provide the following password management capabilities for administrative passwords:

1. Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters: [“!”, “%”, “^”, “(“, “)”, [*“+”, “-“, “[“, “]”, “`”, “~”, “{“, “}”, “|”*]];

Application Note: *The TSF also supports the underscore (_) character for administrative passwords but this was not included in the text of the assignment because the formatting conventions would cause it to be ambiguously represented.*

2. Minimum password length shall settable by the Security Administrator, and support passwords of 15 characters or greater;

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

FIA_UAU_EXT.2.1 The TSF shall provide a local password-based authentication mechanism, [*SSH public-key based authentication mechanism*] to perform administrative user authentication.

6.3.4.3 ***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.

6.3.4.4 ***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;
- [[*display of diagnostic non-TSF environmental data e.g., temperature, fan speed*]]

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.

6.3.5 Class FMT: Security Management

6.3.5.1 ***FMT_MTD.1 Management of TSF Data (for general TSF data)***

FMT_MTD.1.1 The TSF shall restrict the ability to manage the TSF data to the Security Administrators.

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

FMT_SMF.1.1 The TSF shall be capable of performing the following management functions:

- Ability to administer the TOE locally and remotely;
- Ability to update the TOE, and to verify the updates using [digital signature] capability prior to installing those updates;
- [Ability to configure the cryptographic functionality]

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

FMT_SMR.2.1 The TSF shall maintain the roles:

- Authorized Administrator.

Application Note: *The Authorized Administrator role as defined by the NDPP is met through the combination of the 'superuser' role used to manage the MCLI and the Account Administrator (AA) role that is defined for the TL1 interface.*

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:

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

are satisfied.

6.3.6 Class FPT: Protection of the TSF

6.3.6.1 *FPT_APW_EXT.1 Protection of Administrator Passwords*

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

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

6.3.6.2 *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.

6.3.6.3 *FPT_STM.1 Reliable Time Stamps*

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

6.3.6.4 *FPT_TST_EXT.1 TSF Testing*

FPT_TST_EXT.1.1 The TSF shall run a suite of self tests during initial start-up (on power on) to demonstrate the correct operation of the TSF.

6.3.6.5 *FPT_TUD_EXT.1 Trusted Update*

FPT_TUD_EXT.1.1 The TSF shall provide security administrators the ability to query the current version of the TOE firmware/software.

FPT_TUD_EXT.1.2 The TSF shall provide security administrators the ability to initiate updates to TOE firmware/software.

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

6.3.7 Class FTA: TOE Access

6.3.7.1 *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.

Application Note: *'Security Administrator' in this case is considered to be synonymous with Authorized Administrator as defined in FMT_SMR.2.*

6.3.7.2 *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.

Application Note: *'Security Administrator' in this case is considered to be synonymous with Authorized Administrator as defined in FMT_SMR.2.*

6.3.7.3 *FTA_SSL.4 User-initiated Termination*

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

6.3.7.4 *FTA_TAB.1 Default TOE Access Banners*

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.

6.3.8 Class FTP: Trusted Path/Channels

6.3.8.1 *FTP_ITC.1 Inter-TSF Trusted Channel*

FTP_ITC.1.1 The TSF shall use [TLS, SSH] to provide a trusted communication channel between itself and authorized IT entities supporting the following capabilities: audit server, [[*update server*]] that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from disclosure and detection of modification of the channel data.

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

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for [*transfer of audit data, acquisition of TOE updates*].

6.3.8.2 *FTP_TRP.1 Trusted Path*

FTP_TRP.1.1 The TSF shall use [SSH] provide a trusted communication path between itself and 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 detection of modification of the communicated data**.**

FTP_TRP.1.2 The TSF shall permit remote administrators to initiate communication via the trusted path.

FTP_TRP.1.3 The TSF shall require the use of the trusted path for initial administrator authentication and all remote administration actions.

6.4 Statement of Security Functional Requirements Consistency

The Security Functional Requirements included in the ST represent all required SFRs specified in the claimed PP as well as 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 consistent with the claimed PP.

7.1 Class ADV: Development

7.1.1 Basic Functional Specification (ADV_FSP.1)

7.1.1.1 *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.

7.1.1.2 *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.

7.1.1.3 *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.

7.2 Class AGD: Guidance Documents

7.2.1 Operational User Guidance (AGD_OPE.1)

7.2.1.1 ***Developer action elements:***

AGD_OPE.1.1D

The developer shall provide operational user guidance.

7.2.1.2 ***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.

7.2.1.3 *Evaluator action elements:*

AGD_OPE.1.1E

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

7.2.2 Preparative Procedures (AGD_PRE.1)

7.2.2.1 *Developer action elements:*

AGD_PRE.1.1D

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

7.2.2.2 *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.

7.2.2.3 *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.

7.3 Class ALC: Life-cycle Support

7.3.1 Labeling of the TOE (ALC_CMC.1)

7.3.1.1 *Developer action elements:*

ALC_CMC.1.1D

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

7.3.1.2 *Content and presentation elements:*

ALC_CMC.1.1C

The TOE shall be labeled with its unique reference.

7.3.1.3 *Evaluator action elements:*

ALC_CMC.1.1E

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

7.3.2 TOE CM coverage (ALC_CMS.1)

7.3.2.1 *Developer action elements:*

ALC_CMS.1.1D

The developer shall provide a configuration list for the TOE.

7.3.2.2 *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.

7.3.2.3 *Evaluator action elements:*

ALC_CMS.1.1E

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

7.4 Class ATE: Tests

7.4.1 Independent testing - conformance (ATE_IND.1)

7.4.1.1 *Developer action elements:*

ATE_IND.1.1D

The developer shall provide the TOE for testing.

7.4.1.2 *Content and presentation elements:*

ATE_IND.1.1C

The TOE shall be suitable for testing.

7.4.1.3 *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.

7.5 Class AVA: Vulnerability Assessment

7.5.1 Vulnerability Survey (AVA_VAN.1)

7.5.1.1 *Developer action elements:*

AVA_VAN.1.1D

The developer shall provide the TOE for testing.

7.5.1.2 *Content and presentation elements:*

AVA_VAN.1.1C

The TOE shall be suitable for testing.

7.5.1.3 *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, User Data Protection, Identification and Authentication, Security Management, Protection of the TSF, TOE Access and Trusted Path / Channels

8.1 Security Audit

8.1.1 FAU_GEN.1:

The TSF generates audit records of the TOE's behavior. Specifically, the following security-relevant events are audited:

Requirement	Auditable Events
FAU_GEN.1	Start-up and shut-down of the audit functions.
FCS_SSH_EXT.1	Failure to establish an SSH session. Establishment/Termination of an SSH session.
FIA_UIA_EXT.1	All use of the identification and authentication mechanism.
FIA_UAU_EXT.2	All use of the authentication mechanism.
FPT_STM.1	Changes to the time.
FPT_TUD_EXT.1	Initiation of update.
FTA_SSL_EXT.1	Any attempts at unlocking of an interactive session.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.
FTA_SSL.4	The termination of an interactive session.
FTP_ITC.1	Initiation of the trusted channel. Termination of the trusted channel. Failure of the trusted channel functions.
FTP_TRP.1	Initiation of the trusted channel. Termination of the trusted channel. Failures of the trusted path functions.

Table 8-1: Audit Events

Auditing is always functional and thus cannot be disabled or enabled. As a result, the starting up and shutting down of audit functions is synonymous with the startup and shutdown of the TOE. Within each of the audited events listed above, the TOE records at least the date and time of the event, the type of event, the subject's claimed identity, and the outcome (success or failure) of that event. Additional attributes that the TOE records for specific events have been listed in the 'Additional Details' Column of Table 6-2. Date and time is derived from the TOE's hardware clock or optionally for an NTP server.

8.1.2 FAU_GEN.2:

The TOE ensures that each auditable event that is user-initiated includes the identity of the user that performed the function. This is shown in the following sample audit record:

```
2014-07-11T15:06:46.180-04:00 10.88.5.58 409 <21>1 2014-07-11T19:06:44.000+00:00 1-A-CM1
core_nc - - [meta sequenceId="26"] MoName=/NE=VM1/T=SESSION/N=2124338091,
ObjectType=UserSession, ProbableCause=ResponseFromService, Category=Security,
UserAccount=administrator, OpName=LogOn, OpResult=OK, ClientHostName=gax-bkilpatr-
01.atl.ciena.com, ClientInterface=NodeManager, EventData= <P       N="EventData">
```

The ‘UserAccount’ field shows the account name of the user (administrator) that is logging on to the TOE in the event described by this log.

8.1.3 FAU_STG_EXT.1:

The TOE is not an audit server. In the evaluated configuration, the TOE is configured to transmit its collected audit data to a syslog server in the Operational Environment.

Locally, the TOE maintains the security-relevant audit data in two locations on the filesystem, summarized below:

- /rel/<rel-name>/ctm30/<core>/ps/data/AuditTrail: audit log, records all authentication events and management activities performed on the MCLI and TL1 interfaces
- /var/log/secmessages: security syslog, records all events related to user account management

The audit data is stored in up to four files for each audit storage location. Each audit log file stores up to 1,000 records and each security syslog file stores up to 10 MB of data. When storage space is exhausted for either audit storage location, the oldest log file will be overwritten when storage space is exhausted. The TOE does not provide a mechanism to delete the locally-stored audit data.

The TOE uses syslog-ng to persistently transmit audit data from both sources remotely using TCP. This channel is protected using TLS. The local audit storage is used to provide a protection against data loss in the event that the TOE is temporarily unable to communicate with the environmental syslog server. If desired, the same audit data can also be sent to a remote server manually using SFTP (FTP over SSH) using the MCLI.

8.2 Cryptographic Support

8.2.1 FCS_CKM.1:

The TOE implements a NIST SP 800-56A-conformant key generation mechanism for Diffie-Hellman based key establishment schemes. Specifically, the TOE complies with the NIST SP 800-56A key agreement scheme (KAS) primitives that are defined in section 5.6 of the SP. This is used to generate the keys for diffie-hellman-group14-sha1. The TOE also complies with section 6 of NIST SP 800-56B for RSA-based key establishment schemes that are used for TLS communications initiated by the TOE.

8.2.2 FCS_CKM_EXT.4:

The TOE performs key and cryptographic material destruction. The OpenSSL cryptographic module automatically zeroizes sensitive data via API function calls for any data that resides in temporary memory. In each case, the cryptographic data is overwritten in memory with all zeroes when no longer in use. This includes SSH host and session key data that is needed to establish SSH communications as well as TLS key data. In each case, the cryptographic data is overwritten in memory with all zeroes when no longer in use before the memory is freed. Similarly, plaintext password data that is entered by a user as part of the authentication process will be zeroized from temporary memory once the password has been hashed.

Permanently stored SSH keys can be securely erased by an authorized administrator by setting the TOE into maintenance mode. The erasure is done by using the linux ‘shred’ function, with the flags –fn3zv.

This overwrites the stored data on the TOE's ext2 filesystem storage three times with random data followed by a single write with all zeroes.

8.2.3 FCS_COP.1(1):

The TOE provides symmetric encryption and decryption capabilities using AES in CBC mode (128 and 256 bits) as described in NIST SP 800-38A. The TOE provides encryption and decryption in support of TLS and SSH communications. The TOE's AES implementation is validated under CAVP, certificate #3753.

8.2.4 FCS_COP.1(2):

The TOE will provide cryptographic signature services using RSA. RSA is used in support of TLS and SSH communications. All RSA modulus sizes are 2048 bits or larger. The TOE's RSA implementation is validated under CAVP, certificate #1930.

8.2.5 FCS_COP.1(3):

The TOE provides cryptographic hashing services using SHA-1 and SHA-256 as specified in FIPS Pub 180-3 “Secure Hash Standard”. The TOE uses cryptographic hashing services in support of SSH key establishment as well as session establishment for TLS and SSH communications. The TOE's SHS implementation is validated under CAVP, certificate #3124.

8.2.6 FCS_COP.1(4):

The TOE provides keyed-hash message authentication services using HMAC-SHA-1 and HMAC-SHA-256, as specified in FIPS Pub 198-1,"The Keyed-Hash Message Authentication Code,” and FIPS 180-3, “Secure Hash Standard”. All key sizes relative to block sizes are supported by the HMAC implementation. The TOE's HMAC implementation is validated under CAVP, certificate #2456.

8.2.7 FCS_RBG_EXT.1:

The TOE implements a NIST-approved deterministic random bit generator (DRBG). The DRBG used by the TOE is a NIST Special Publication 800-90 Hash_DRBG. The TOE models provide a hardware-based entropy source as described in the proprietary entropy specification. The DRBG is seeded with a minimum of 256 bits of entropy so that it is sufficient to ensure full entropy for 256-bit keys, which are the largest keys generated by the TSF. The TOE's DRBG implementation is validated under CAVP, certificate #1029.

8.2.8 FCS_SSH_EXT.1:

The TOE implements SSHv2 for remote MCLI sessions that complies with RFCs 4251, 4252, 4253, and 4254. There is no SSHv1 implementation on the TOE.

SSH connections will be dropped if the TOE receives a packet larger than 32,000 bytes. Large packets are detected by the SSH implementation and dropped by the SSH process.

The TOE implementation of SSHv2 supports RSA signature verification for authentication in addition to password-based authentication.

The TOE implementation of SSHv2 supports AES-CBC-128 and AES-CBC-256 for its transport algorithms and RSA as it’s only supported public key algorithm. Data integrity is assured using either HMAC-SHA-1 or HMAC-SHA-256. The allowed key exchange method is diffie-hellman-group14-sha1.

8.2.9 FCS_TLS_EXT.1:

The TOE can initiate outbound TLS connections for secure communications with an environmental syslog server. TLS 1.2 (RFC 5246) is the supported TLS version. The TOE supports the following TLS ciphersuites:

TLS_RSA_WITH_AES_128_CBC_SHA

TLS_RSA_WITH_AES_256_CBC_SHA

TLS_RSA_WITH_AES_128_CBC_SHA256

TLS_RSA_WITH_AES_256_CBC_SHA256

8.3 User Data Protection

8.3.1 FDP_RIP.2:

The TOE ensures that packets transmitted from the TOE do not contain residual information from previous packets. Packets that are not the required length use zeroes for padding. The TOE ensures that packets transmitted from the product do not contain any residual information by zeroizing the data upon allocation of memory. This ensures that if a new packet reuses the same memory location as a previous packet, the location is zeroized first before the new packet is constructed. In certain instances the TOE may also zeroize data immediately after the memory is deallocated; which would occur in addition to the zeroization function that occurs when memory is allocated for a packet.

8.4 Identification and Authentication

8.4.1 FIA_PMG_EXT.1:

The TOE supports the local definition of users with corresponding passwords. The passwords can be composed of any combination of upper and lower case letters, numbers, and special characters. The supported special characters include “!”, “%”, “^”, “(“, “)”, “+”, “-“, “[“, “]”, “`”, “~”, “{“, “}”, “|”, and “_”. Passwords can be between 6 and 16 characters in length, and an authorized administrator has the ability to set the minimum length that is permitted to any value between 6 and 16. In the evaluated configuration passwords must be set to 15 characters or greater.

8.4.2 FIA_UAU_EXT.2:

The TOE requires the use of locally-defined authentication credentials. Users are not allowed to perform any security-relevant functions on the TOE without first being successfully identified and authenticated by the TOE’s authentication method. Some environmental diagnostic information is displayed such as fan speed and temperature, but this data is only visible at the local serial console, cannot be manipulated, and does not have security relevance. At initial login in the administrative user is prompted to provide a username. After the user provides the username, the user is prompted to provide the administrative password associated with the user account. Alternatively, if configured to do so, the user can use public-key authentication to log in to the MCLI remotely using SSH. The TOE then either grants administrative access (if the combination of username and credential are correct) or indicates that the login was unsuccessful. The MCLI requires a separate superuser account that cannot be the same as an account that is used to access the TL1 interface. However, the TOE stores username and password hash data in the same local storage for both the MCLI and TL1 interfaces.

8.4.3 FIA_UAU.7:

When a user enters their password at the local console, the password characters entered by the user are not echoed back to them. When accessing the TOE remotely, both of these logical interfaces use SSH so the specific method of obfuscation is dependent on the SSH client used by the management workstation.

8.4.4 FIA_UIA_EXT.1:

See FIA_UAU_EXT.2 above.

8.5 Security Management

8.5.1 FMT_MTD.1:

The TOE restricts access to the management functions to an authorized administrator. The product provides five administrative roles on its TL1 interface: Account Administrator (AA), Termination Point Provisioner (TP), Connection Provisioner (CP), Troubleshooter (TS), and Operator (O). Each administrative role is given a fixed set of privileges. Of these five roles, only the AA role has the ability to manage functions that are relevant to the TOE as defined by the NDPP. The remaining functions pertain to the management of switching functions that are outside the scope of the NDPP. The TOE also provides a CLI (referred to as the MCLI) for many security-relevant features, typically those that are not managed frequently (such as configuring communications with remote audit and NTP servers). The MCLI defines a superuser role that is separate from the roles defined for the TL1 interface, and is the only role that is defined for the MCLI. For the purposes of the TSF, an authorized administrator is considered to be either the MCLI superuser or an AA using the TL1 interface, depending on the functions being performed.

8.5.2 FMT_SMF.1:

The TOE provides all the capabilities necessary to securely manage the TSF. The TOE includes a menu-driven MCLI and a TL1 interface (which is also technically a command-line interface but not referred to as such in order to avoid confusion with the menu-driven MCLI). The MCLI primarily includes functions that are performed when the TOE is initially deployed or that only need to be performed periodically, whereas the functions provided by TL1 are more associated with day-to-day operations. The following table describes the management functions provided by the TOE along with the interface an authorized administrator will use to perform these functions:

Management Function	Interface Used
Creation of user accounts and assignment to administrative roles	TL1
Specification of maximum idle time for an administrative session before it is terminated	TL1
Configuration of minimum password length	TL1
Configuration and manual transfer of audit data to remote storage location	MCLI
Manual setting of system time	TL1
Configuration of NTP server connection	MCLI

Management of cryptographic functions	MCLI
Configuration of banner text	TL1
Initiation of system software/firmware update	MCLI

Table 8-2: TSF Management Functions

The MCLI can be administered locally via serial port and remotely using SSH whereas the TL1 interface can be accessed via SSH only. If administering the TOE locally via TL1 is desired, the management workstation should be placed on the same local network as the TOE.

8.5.3 FMT_SMR.2:

The TOE maintains several administrative roles, each of which has a fixed set of allowed operations. Of the roles defined by the product, the Account Administrator (AA) and superuser roles are those that apply to the management of the TSF. These roles serve as the Security Administrator or Authorized Administrator for their respective interfaces. An administrator may only have one role assigned to their account. The superuser role is defined only for the MCLI interface, and it is the only role that is authorized to use this interface. The AA role is defined only for the TL1 interface. If an individual must perform functions that require the use of both MCLI and TL1 interfaces, it is necessary for that individual to have two separate administrative accounts. This is done because there is typically only one superuser account which is rarely used after initial configuration.

8.6 Protection of the TSF

8.6.1 FPT_APW_EXT.1:

Administrator passwords are not stored by the TOE. All administrative passwords are hashed using SHA-256 and the hash is what is stored by the TOE. There is no function provided by the TOE to display a password value in plaintext.

8.6.2 FPT_SKP_EXT.1:

The TOE does not provide a mechanism to view secret keys and key material. Public key data that is stored on the TOE can be viewed by an authorized administrator. Key data that is resident in volatile memory cannot be accessed by an administrative command. Any persistent key data is stored in the underlying filesystem of the OS on internal flash memory. The TOE's management interfaces do not provide any direct access to the file system so there is no administrative method of accessing this data.

8.6.3 FPT_STM.1:

The TOE provides a source of date and time information, used in audit timestamps and in determining whether an administrative session has gone active. The clock function is reliant on the system clock provided by the underlying hardware. The TOE can optionally be set to receive time from one or more NTP servers. If using the local clock, an authorized administrator has the ability to manually set the time using the MCLI.

8.6.4 FPT_TST_EXT.1:

The TOE runs a series of self-tests during initial start-up to verify its correct operation. As part of the startup of the TOE, the OpenSSL cryptographic module will perform a series of known answer tests to verify the correct functionality of the cryptographic functions as well as fingerprint and SHA file checksums to validate its own integrity. Independent of the cryptographic module, the integrity of the software image itself is also validated against a known hash to ensure its integrity. In the event that a cryptographic self-test fails, the TOE will attempt to reboot itself. These tests are sufficient to ensure that the TOE is functioning in the manner that is expected because they will detect a modified software image or improperly deployed cryptographic module.

8.6.5 FPT_TUD_EXT.1:

The TOE provides the ability for an authorized administrator to update its software. The TOE has an SFTP client that is used to retrieve software updates from an SFTP server. This can be a server maintained by Ciena or one maintained by the organization operating the TOE, in which case updates are shipped on read-only physical media when made available by Ciena and then loaded onto the SFTP server in the Operational Environment. Updates are both digitally signed and hashed, but the hash information is used only for internal verification and not made public. The digital signature is a 2048-bit RSA signature that is provided by Entrust. Once the update has been uploaded to the TOE, the digital signature of the software upgrade is verified. If the digital signature verification fails, the upgrade process will stop and the downloaded software release will be flushed from the device's temporary memory.

8.7 TOE Access

8.7.1 FTA_SSL_EXT.1:

The Administrator can configure maximum inactivity times for both local and remote administrative sessions. When a session is inactive for the configured period of time the TOE will terminate the session, requiring the administrator to log in again to establish a new session when needed.

8.7.2 FTA_SSL.3:

The TOE will terminate a TL1 or MCLI session after an administrator-defined period of inactivity. The idle timeout value is set for each individual user account as opposed to being globally defined for all users. This is specified using the 'tmout' parameter when the user is created or modified using the TL1 interface. By default, a user account will be logged out if idle for 60 minutes, but the value can be set to anything between 1 and 999 minutes.

8.7.3 FTA_SSL.4:

The TOE provides the ability for administrators to manually terminate their own sessions. TL1 uses the CANC-USER command and MCLI uses menu choice '10'. These commands apply to both local and remote usage. Additionally, when managing the TOE remotely, the terminal application used on the management workstation will typically terminate the SSH session if the application itself is closed.

8.7.4 FTA_TAB.1:

The TOE displays a configurable warning banner on the local console prior to an administrator supplying their authentication credentials. The TOE also displays a configurable warning banner on both local and remote consoles prior to allowing the input of authentication credentials. The warning banner is configured by an authorized administrator.

8.8 Trusted Path/Channels

8.8.1 FTP_ITC.1:

The TOE provides the ability to secure sensitive data in transit to and from the Operational Environment. In the evaluated configuration, the TOE is configured to transmit audit data to a remote syslog server using TLS and to a remote file server using SFTP, which uses SSH to secure FTP communications. Updates to the TOE software and manually transferring audit data are securely transmitted using SFTP. In both cases, the TOE relies on the CAVP-validated cryptographic algorithm implementation used to establish these trusted channels.

8.8.2 FTP_TRP.1:

All remote administrative communications, regardless of which logical interface they originate from, take place over a secure encrypted SSHv2 session. The TOE uses OpenSSH to perform SSH functions, which in turn relies on the CAVP-validated cryptographic algorithm implementation to provide the cryptographic algorithm services used to perform SSH.