Cisco Prime Infrastructure

Security Target

Version 1.3

April 25th, 2018

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Table of Contents

List of Tables

List of Figures

List of Acronyms

The following acronyms and abbreviations are common and may be used in this Security Target:

Table 1 Acronyms

Acronyms / Abbreviations	Definition
AAA	Administration, Authorization, and Accounting
ACL	Access Control Lists
AES	Advanced Encryption Standard
AES-CCM	AES Counter with CBC-MAC
CC	Common Criteria for Information Technology Security Evaluation
CEM	Common Evaluation Methodology for Information Technology Security
CM	Configuration Management
DHCP	Dynamic Host Configuration Protocol
EAL	Evaluation Assurance Level
EAP	Extensible Authentication Protocol
EAPoL	Extensible Authentication Protocol (EAP) over LAN
ESP	Encapsulating Security Payload
GE	Gigabit Ethernet port
HTTP	Hyper-Text Transport Protocol
HTTPS	Hyper-Text Transport Protocol Secure
ICMP	Internet Control Message Protocol
IT	Information Technology
KCK	Key Confirmation Key
KEK	Key Encryption Key
MIC	Message Integrity Check
NDcPP	collaborative Network Device Protection Profile
OS	Operating System
PoE	Power over Ethernet
PP	Protection Profile
PTK	Pairwise Transient Key
RSN	Robust Security Network
SA	Security Association
SFP	Small–form-factor pluggable port
SHS	Secure Hash Standard
SSHv2	Secure Shell (version 2)
ST	Security Target
TCP	Transport Control Protocol
TOE	Target of Evaluation
TSC	TSF Scope of Control
TSF	TOE Security Function
TSP	TOE Security Policy
UDP	User datagram protocol
WAN	Wide Area Network

DOCUMENT INTRODUCTION

Prepared By:
Cisco Systems, Inc.
170 West Tasman Dr.
San Jose, CA 95134

This document provides the basis for an evaluation of a specific Target of Evaluation (TOE), the Cisco Prime Infrastructure (PI). This Security Target (ST) defines a set of assumptions about the aspects of the environment, a list of threats that the product intends to counter, a set of security objectives, a set of security requirements, and the IT security functions provided by the TOE which meet the set of requirements. Administrators of the TOE will be referred to as administrators, Authorized Administrators, TOE administrators, semi-privileged, privileged administrators, and security administrators in this document.

REVISION HISTORY

Rev	Date	Description
1.3	April 25th, 2018	Final Version

1 SECURITY TARGET INTRODUCTION

The Security Target contains the following sections:

- Security Target Introduction [Section 1]
- Conformance Claims [Section 2]
- Security Problem Definition [Section 3]
- Security Objectives [Section 4]
- IT Security Requirements [Section 5]
- TOE Summary Specification [Section 6]
- Supplemental TOE Summary Specification Information [Section 7]
- References [Section 8]

The structure and content of this ST comply with the requirements specified in the Common Criteria (CC), Part 1, Annex A, and Part 2.

1.1 ST and TOE Reference

This section provides information needed to identify and control this ST and its TOE.

Table 2: ST and TOE Identification

Name	Description
ST Title	Cisco Prime Infrastructure
ST Version	1.3
Publication Date	April 25th, 2018
Vendor and ST Author	Cisco Systems, Inc.
TOE Reference	Cisco Prime Infrastructure
TOE Hardware Models	Cisco Prime Infrastructure Physical Appliance (Gen 2)
TOE Software	Cisco Prime Infrastructure 3.2-FIPS Physical Appliance ISO image with PI-patch-fips-1
Keywords	Network Management

1.2 TOE Overview

The Cisco Prime Infrastructure TOE is a purpose-built network device that supports management of an organization's entire network infrastructure from one graphical interface. Cisco Prime Infrastructure provides a single integrated solution for comprehensive lifecycle management of the organization's infrastructure including network equipment, servers and virtual machines. Cisco Prime Infrastructure supports management of devices in the infrastructure with IPv4 or IPV6 addressing and uses the industry-standard SNMP protocol for communication.

The focus of the evaluation is on the baseline network device requirements from the NDcPP.

TOE Product Type

The Cisco Prime Infrastructure TOE is a network management device that provides network administrators with a single solution for provisioning, monitoring, optimizing, and troubleshooting both wired and wireless devices.

Required non-TOE Hardware/ Software/ Firmware

The TOE requires the following hardware, software, and firmware in its environment when the TOE is configured in its evaluated configuration:

Table 3: Required IT Environment Components

Component	Usage/Purpose Description
Management Workstation	This includes: o A Mac or Windows laptop or desktop with one of the following web browsers supporting TLS v1.2: ▪ Google Chrome 59 or later ▪ Microsoft Internet Explorer 11 (No plug-ins are required.) ▪ Mozilla Firefox ESR 52 ▪ Mozilla Firefox 56 or later o A Mac, UNIX, or Windows laptop or desktop with a SSH v2 client
Local Console	This includes any IT Environment Console that is directly connected to the TOE component via the Serial Console Port and supports a baud rate of 9600 bits per second.
Syslog Server	This includes any syslog server that can be accessed over TLS v1.2 with the supported ciphersuites.

1.3 TOE DESCRIPTION

The Cisco Prime Infrastructure 3.2-FIPS TOE is a network management product comprised of hardware and software. The focus of the evaluation is on the baseline network device requirements from the NDcPP which are briefly described below.

The TOE provides encryption for transmitting sensitive data from itself and an endpoint interacting with the TOE. Additionally, to assure there is a trusted means for administrators and peer devices to communicate with the TOE, X.509 certificate authentication and validation is provided. The TOE's support for encryption has been CAVP tested to assure cryptographic algorithms have been implemented correctly.

A Security Administrator role is provided along with a set of security management functions. Secure remote administration to the Web GUI and CLI is protected with HTTPS and SSHv2, respectively. Local and remote sessions are monitored for inactivity and are terminated when a threshold time period is reached.

The TOE protects critical security data such as keys and passwords and provides self-tests that monitor continued correct operation. In addition the TOE provides trusted methods for software updates.

Lastly, to assure that information exists which will allow Security Administrators to discover intentional and unintentional issues with the configuration and/or operation of the Cisco Prime Infrastructure, the TOE provides an auditing function.

An image of the TOE is provided in figure 1 below:

Figure 1

1.4 TOE Evaluated Configuration

The TOE deployed in its evaluated configuration consists of one physical device as specified in section 1.5 below. The TOE physical boundary is depicted in blue shading.

The operational environment of the TOE will include an audit (syslog) server. The TOE can be administered interactively using a local console connection (CLI), or remotely over HTTPS (GUI) or SSH v2.

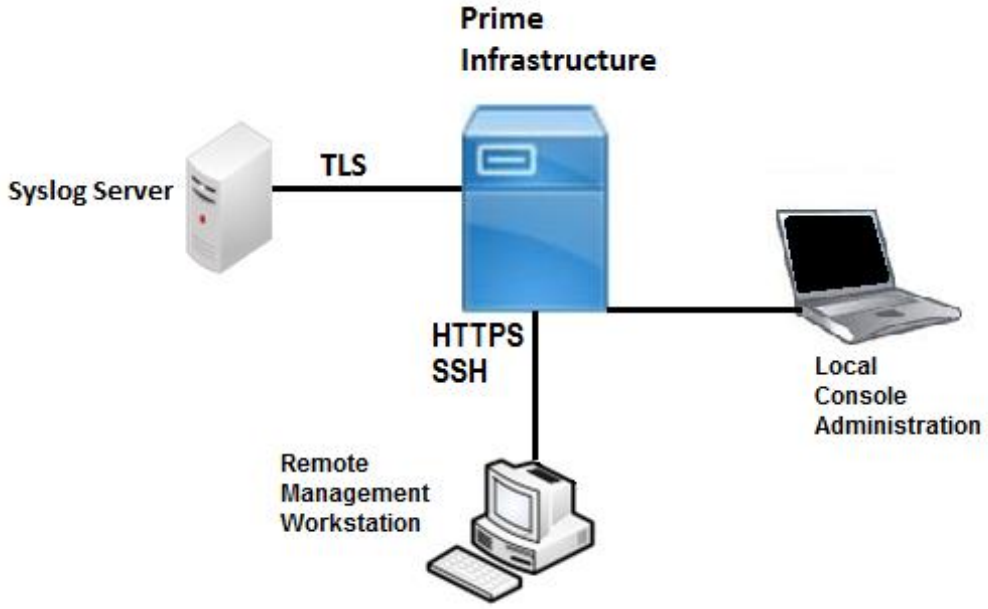

Figure 2 TOE and Environment

1.5 Physical Scope of the TOE

The physical boundary of the Cisco Prime Infrastructure 3.2-FIPS TOE is described in Table 4 below.

Table 4 TOE Boundary

Hardware	Cisco Prime Infrastructure Physical Appliance (Gen 2) o Product ID: PI-UCS-APL-K9 o Hardware model: Cisco UCS-C220–M4 o Processor: Dual Intel® Xeon E5-2650 v3 @2.30GHz 8 core processor o Memory: 64 GB o Hard Disk: 4 x 900GB RAID10 o NIC: Integrated dual-port Gigabit Ethernet
Software	Cisco Prime Infrastructure 3.2-FIPS Physical Appliance ISO image with PI-patch-fips-1

1.6 Logical Scope of the TOE

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. Identification and Authentication
4. Security Management
5. Protection of the TSF
6. TOE Access
7. Trusted Path/Channels

These features are described in more detail in the subsections below.

Security Audit

Auditing allows Security Administrators to discover intentional and unintentional issues with the TOE's configuration and/or operation. Auditing of administrative activities provides information that may be used to hasten corrective action should the system be configured incorrectly. Security audit data can also provide an indication of failure of critical portions of the TOE (e.g. a communication channel failure or anomalous activity (e.g. establishment of an administrative session at a suspicious time, repeated failures to establish sessions or authenticate to the TOE) of a suspicious nature.

The TOE provides extensive capabilities to generate audit data targeted at detecting such activity. The TOE generates an audit record for each auditable event. Each security relevant audit event has the date, timestamp, event description, and subject identity. The administrator configures auditable events, performs back-up operations, and manages audit data storage. The TOE provides the administrator with a circular audit trail or a configurable audit trail threshold to track the storage capacity of the audit trail. Audit logs are transmitted to an external audit server over an encrypted channel.

Cryptographic Support

The TOE implements cryptography and algorithms that has been CAVP tested. This includes key generation and random bit generation, key establishment methods, key destruction, and the various types of cryptographic operations to provide AES encryption/decryption, signature verification, hash generation, and keyed hash generation.

Identification and Authentication

The TOE performs two types of authentication to provide a trusted means for Security Administrators and remote endpoints to interact with itself: X.509v3 certificate-based authentication for remote devices and password-based authentication for Security Administrators. Device-level authentication allows the TOE to establish a secure communication channel with a remote endpoints.

Security Administrators have the ability to compose strong passwords (15 characters or greater), which are stored in a hashed form.

Security Management

The TOE provides secure remote administrative interface and a local interface to perform security management functions. This includes ability to configure cryptographic functionality; an access banner containing an advisory notice and consent warning message; a session inactivity time before session termination as well as an ability to update its software.

The TOE provides a Security Administrator role and only the Security Administrator can perform security management functions.

Protection of the TSF

The TOE protects critical security data including keys and passwords against tampering by untrusted subjects. The TOE also provides reliable timestamps to support accurate audit records.

The TOE provides self-tests to ensure it is operating correctly, including the ability to detect software integrity failures. Additionally, the TOE provides a trusted software update and verification function to assure software updates are from Cisco Systems, Inc.

TOE Access

The TOE monitors both local and remote admin sessions for inactivity and terminates when a threshold time period is reached. Once a session has been terminated the TOE requires the user to re-authenticate.

The TOE also displays a Security Administrator specified advisory notice and consent warning message prior to initiating identification and authentication for each administrative user.

Trusted path/Channels

The TOE provides encryption (protection from disclosure and detection of modification) for communication paths between itself and remote endpoints.
In addition the TOE provides two-way authentication of each endpoint in a cryptographically secure manner, meaning that even if there was a malicious attacker between the two endpoints, any attempt to represent themselves to either endpoint of the communications path as the other communicating party would be detected.

1.7 Excluded and Functionality Not Covered

The following functionality is excluded or not covered in the CC evaluation.

Table 5: Excluded and Functionality Not Covered

Functionality	Rationale
Management of remote network devices using SNMP	This feature is available in the evaluated configuration but not covered by security functional requirements in the NDcPP.
The virtual KVM of the Cisco Integrated Management Controller (CIMC) interface SoL (Serial over Lan) through CIMC	Remote TOE management using CIMC is not permitted and is excluded in the evaluated configuration and will be disabled by configuration.

2 CONFORMANCE CLAIMS

2.1 Common Criteria Conformance Claim

The TOE and ST are compliant with the Common Criteria (CC) Version 3.1, Revision 4, dated: September 2012. For a listing of Assurance Requirements claimed see section 5.4.

The TOE and ST are CC Part 2 extended and CC Part 3 conformant.

2.2 Protection Profile Conformance

The TOE and ST are conformant with the Protection Profiles as listed in Table 6 below:

Table 6: Protection Profiles

Protection Profile	Version	Date
collaborative Protection Profile for Network Devices (NDcPP)	1.0	February 27, 2015

This ST applies the following NIAP Technical Decisions:

- TD0090: NIT Technical Decision for FMT_SMF.1.1 Requirement in NDcPP
- TD0094: NIT Technical Decision for validating a published hash in NDcPP
- TD0095: NIT Technical Interpretations regarding audit, random bit generation, and entropy in NDcPP
- TD0111: NIT Technical Decision for third party libraries and FCS_CKM.1 in NDcPP and FWcPP
- TD0112: NIT Technical Decision for TLS testing in the NDcPP v1.0 and FW cPP v1.0.
- TD0114: NIT Technical Decision for Re-Use of FIPS test results in NDcPP and FWcPP
- TD0116: NIT Technical Decision for a Typo in reference to RSASSA-PKCS1v1_5 in NDcPP and FWcPP
- TD0117: NIT Technical Decision for FIA_X509_EXT.1.1 Requirement in NDcPP
- TD0125: NIT Technical Decision for Checking validity of peer certificates for HTTPS servers
- TD0126: NIT Technical Decision for TLS Mutual Authentication
- TD0130: NIT Technical Decision for Requirements for Destruction of Cryptographic Keys
- TD0143: Failure testing for TLS session establishment in NDcPP and FWcPP
- TD0150: NIT Technical Decision for Removal of SSH re-key audit events in the NDcPP v1.0 and FW cPP v1.0
- TD0151: NIT Technical Decision for FCS_TLSS_EXT Testing - Issue 1 in NDcPP v1.0
- TD0152: NIT Technical Decision for Reference identifiers for TLS in the NDcPP v1.0 and FW cPP v1.0
- TD0153: NIT Technical Decision for Auditing of NTP Time Changes in the NDcPP v1.0 and FW cPP v1.0

- TD0154: NIT Technical Decision for Versions of TOE Software in the NDcPP v1.0 and FW cPP v1.0
- TD0155: NIT Technical Decision for TLSS tests using ECDHE in the NDcPP v1.0
- TD0156: NIT Technical Decision for SSL/TLS Version Testing in the NDcPP v1.0 and FW cPP v1.0
- TD0164: NIT Technical Decision for Negative testing for additional ciphers for SSH
- TD0165: NIT Technical Decision for Sending the ServerKeyExchange message when using RSA
- TD0167: NIT Technical Decision for Testing SSH 2^28 packets
- TD0168: NIT Technical Decision for Mandatory requirement for CSR generation
- TD0169: NIT Technical Decision for Compliance to RFC5759 and RFC5280 for using CRLs
- TD0170: NIT Technical Decision for SNMPv3 Support
- TD0181: NIT Technical Decision for Self-testing of integrity of firmware and software.
- TD0183: NIT Technical Decision for Use of the Supporting Document
- TD0185: NIT Technical Decision for Channel for Secure Update.
- TD0187: NIT Technical Decision for Clarifying FIA_X509_EXT.1 test 1
- TD0188: NIT Technical Decision for Optional use of X.509 certificates for digital signatures
- TD0189: NIT Technical Decision for SSH Server Encryption Algorithms
- TD0191: NIT Technical Decision for Using secp521r1 for TLS communication
- TD0199: NIT Technical Decision for Elliptic Curves for Signatures
- TD0201: NIT Technical Decision for Use of intermediate CA certificates and certificate hierarchy depth
- TD0226: NIT Technical Decision for TLS Encryption Algorithms
- TD0227: NIT Technical Decision for TOE acting as a TLS Client and RSA key generation
- TD0228: NIT Technical Decision for CA certificates - basicConstraints validation
- TD0235: NIT Technical Decision adding DH group 14 to the selection in FCS_CKM.2
- TD0255: NIT Technical Decision for TLS Server Tests - Issue 3: Verification of application of encryption
- TD0256: NIT Technical Decision for Handling of TLS connections with and without mutual authentication
- TD0257: NIT Technical Decision for Updating FCS_DTLSC_EXT.x.2/FCS_TLSC_EXT.x.2 Tests 1-4
- TD0281: NIT Technical Decision for Testing both thresholds for SSH rekey
- TD0289: NIT technical decision for FCS_TLSC_EXT.x.1 Test 5e
- TD0290: NIT technical decision for physical interruption of trusted path/channel.

This following NIAP Technical Decisions do **not** apply to this ST:

- TD0096: NIT Technical Interpretation regarding Virtualization

- TD0113: NIT Technical Decision for testing and trusted updates in the NDcPP v1.0 and FW cPP v1.0
- TD0115: NIT Technical Decision for Transport mode and tunnel mode in IPsec communication in NDcPP and FWcPP
- TD0160: NIT Technical Decision for Transport mode and tunnel mode in IPSEC communications
- TD0182: NIT Technical Decision for Handling of X.509 certificates related to ssh-rsa and remote comms.
- TD0184: NIT Technical Decision for Mandatory use of X.509 certificates
- TD0186: NIT Technical Decision for Applicability of X.509 certificate testing to IPsec
- TD0200: NIT Technical Decision for Password authentication for SSH clients
- TD0223: NIT Technical Decision for "Expected" vs "unexpected" DNs for IPsec Communications
- TD0224: NIT Technical Decision Making DH Group 14 optional in FCS_IPSEC_EXT.1.11
- TD0225: NIT Technical Decision for Make CBC cipher suites optional in IPsec:
- TD0262: NIT Technical Decision for TLS server testing - Empty Certificate Authorities list
- TD0291: NIT technical decision for DH14 and FCS_CKM.1

2.3 Protection Profile Conformance Claim Rationale

TOE Appropriateness

The TOE provides all of the functionality at a level of security commensurate with that identified in the U.S. Government Protection Profile:

- collaborative Protection Profile for Network Devices, Version 1.0

TOE Security Problem Definition Consistency

The Assumptions, Threats, and Organization Security Policies included in the Security Target represent the Assumptions, Threats, and Organization Security Policies specified in the collaborative Protection Profile for Network Devices 1.0 for which conformance is claimed verbatim. All concepts covered in the Protection Profile Security Problem Definition are included in the Security Target Statement of Security Objectives Consistency.

The Security Objectives included in the Security Target represent the Security Objectives specified in the collaborative Protection Profile for Network Devices 1.0 for which conformance is claimed verbatim. All concepts covered in the Protection Profile's Statement of Security Objectives are included in the Security Target.

Statement of Security Requirements Consistency

The Security Functional Requirements included in the Security Target represent the Security Functional Requirements specified in the collaborative Protection Profile for Network Devices 1.0 for which conformance is claimed verbatim. All concepts covered the Protection Profile's Statement of Security Requirements are included in the Security Target. Additionally, the Security Assurance Requirements included in the Security Target are identical to the Security Assurance Requirements included in the claimed Protection Profiles.

3 SECURITY PROBLEM DEFINITION

This chapter identifies the following:

- Significant assumptions about the TOE's operational environment.
- IT related threats to the organization countered by the TOE.
- Environmental threats requiring controls to provide sufficient protection.
- Organizational security policies for the TOE as appropriate.

This document identifies assumptions as A.assumption with "assumption" specifying a unique name. Threats are identified as T.threat with "threat" specifying a unique name. Organizational Security Policies (OSPs) are identified as P.osp with "osp" specifying a unique name.

3.1 Assumptions

The specific conditions listed in the following subsections are assumed to exist in the TOE's environment. These assumptions include both practical realities in the development of the TOE security requirements and the essential environmental conditions on the use of the TOE.

Table 7 TOE Assumptions

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 and/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 will not include any requirements on physical tamper protection or other physical attack mitigations. The cPP will 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.
A.LIMITED_FUNCTIONALITY	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 computing platform for general purpose applications (unrelated to networking functionality).
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 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 being appropriately trained, following policy, and adhering to guidance documentation. Administrators are trusted to

Assumption	Assumption Definition
	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.
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.

3.2 Threats

The following table lists the threats addressed by the TOE and the IT Environment. The assumed level of expertise of the attacker for all the threats identified below is Enhanced-Basic.

Table 8 Threats

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 tunneling 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.

Threat	Threat Definition
T.WEAK_AUTHENTICATION_ENDPOINTS	Threat agents may take advantage of secure protocols that use weak methods to authenticate the endpoints – e.g., shared password that is guessable or transported as plaintext. The consequences are the same as a poorly designed protocol, the attacker could masquerade as the administrator or another device, and the attacker could insert themselves into the network stream and perform a man-in-the-middle attack. The result is the critical network traffic is exposed and there could be a loss of confidentiality and integrity, and potentially the network device itself could be compromised.
T.UPDATE_COMPROMISE	Threat agents may attempt to provide a compromised update of the software or firmware which undermines the security functionality of the device. Non-validated updates or updates validated using non-secure or weak cryptography leave the update firmware vulnerable to surreptitious alteration.
T.UNDETECTED_ACTIVITY	Threat agents may attempt to access, change, and/or modify the security functionality of the network device without administrator awareness. This could result in the attacker finding an avenue (e.g., misconfiguration, flaw in the product) to compromise the device and the administrator would have no knowledge that the device has been compromised.
T.SECURITY_FUNCTIONALITY_COMPROMISE	Threat agents may compromise credentials and device data enabling continued access to the network device and its critical data. The compromise of credentials include 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	A component of the network device may fail during start-up or during operations causing a compromise or failure in the security functionality of the network device, leaving the device susceptible to attackers.

3.3 Organizational Security Policies

The following table lists the Organizational Security Policies imposed by an organization to address its security needs.

Table 9 Organizational Security Policies

Policy Name	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.

4 SECURITY OBJECTIVES

This Chapter identifies the security objectives of the TOE and the IT Environment. The security objectives identify the responsibilities of the TOE and the TOE's IT environment in meeting the security needs.

- This document identifies objectives of the TOE as O.objective with objective specifying a unique name. Objectives that apply to the IT environment are designated as OE.objective with objective specifying a unique name.

4.1 Security Objectives for the Environment

All of the assumptions stated in section 3.1 are considered to be security objectives for the environment. The following are the Protection Profile non-IT security objectives, which, in addition to those assumptions, are to be satisfied without imposing technical requirements on the TOE. That is, they will not require the implementation of functions in the TOE hardware and/or software. Thus, they will be satisfied largely through application of procedural or administrative measures.

Table 10 Security Objectives for the Environment

Environment Security Objective	IT Environment Security 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.
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	TOE Administrators are trusted to follow and apply all administrator guidance in a trusted manner.
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.

5 SECURITY REQUIREMENTS

This section identifies the Security Functional Requirements for the TOE. The Security Functional Requirements in this section are derived from [CC_PART2], [NDcPP], and ND cPP interpretations.

5.1 Conventions

The CC defines operations on Security Functional Requirements: assignments, selections, assignments within selections and refinements. This document uses the following font conventions to identify the operations defined by the CC:

- Assignment: Indicated with *italicized* text;
- Refinement made by PP author: Indicated with **bold text** and ~~strikethroughs~~, if necessary;
- Selection: Indicated with underlined text;
- Assignment within a Selection: Indicated with *italicized and underlined text*;
- Iteration: Indicated by appending the iteration number in parenthesis, e.g., (1), (2), (3) and/or by adding a string starting with "/"..
- Where operations were completed in the NDcPP itself, the formatting used in the NDcPP has been retained.
- Formatting used in NDcPP that is inconsistent with the listed conventions has not being retained in the ST.

Explicitly stated SFRs are identified by having a label 'EXT' after the requirement name for TOE SFRs. Formatting conventions outside of operations and iterations matches the formatting specified within the NDcPP.

5.2 TOE Security Functional Requirements

This section identifies the Security Functional Requirements for the TOE. The TOE Security Functional Requirements that appear in the following table are described in more detail in the following subsections.

Table 11 Security Functional Requirements

Class Name	Component Identification	Component Name
Security Audit	FAU_GEN.1	Audit data generation
	FAU_GEN.2	User Identity Association
	FAU_STG.1	Protected Audit Trail Storage
	FAU_STG_EXT.1	Protected Audit Event Storage
Cryptographic Support	FCS_CKM.1	Cryptographic Key Generation (for asymmetric keys)
	FCS_CKM.2	Cryptographic Key Establishment (Refined)
	FCS_CKM.4	Cryptographic Key Destruction
	FCS_COP.1(1)	Cryptographic Operation (AES Data Encryption/Decryption)
	FCS_COP.1(2)	Cryptographic Operation (Signature Generation and

Class Name	Component Identification	Component Name
		Verification)
	FCS_COP.1(3)	Cryptographic Operation (Hash Algorithm)
	FCS_COP.1(4)	Cryptographic Operation (Keyed Hash Algorithm)
	FCS_RBG_EXT.1	Random Bit Generation
	FCS_HTTPS_EXT.1	HTTPS Protocol
	FCS_TLSS_EXT.1	TLS Server Protocol
	FCS_TLSC_EXT.1	TLS Client Protocol
	FCS_SSHS_EXT.1	SSH Server Protocol
Identification and Authentication	FIA_PMG_EXT.1	Password Management
	FIA_UIA_EXT.1(1) FIA_UIA_EXT.1(2)	User Identification and Authentication
	FIA_UAU_EXT.2	Password-based Authentication Mechanism
	FIA_UAU.7	Protected Authentication Feedback
	FIA_X509_EXT.1	X.509 Certificate Validation
	FIA_X509_EXT.2	X.509 Certificate Authentication
	FIA_X509_EXT.3	X.509 Certificate Requests
Security Management	FMT_MOF.1(1)/TrustedUpdate	Management of security functions behaviour
	FMT_MOF.1(1)/Audit	Management of security functions behaviour
	FMT_MOF.1(2)/Audit	Management of security functions behaviour
	FMT_MOF.1(1)/AdminAct	Management of security functions behaviour
	FMT_MOF.1(2)/AdminAct	Management of security functions behaviour
	FMT_MTD.1	Management of TSF Data
	FMT_MTD.1/AdminAct	Management of TSF Data
	FMT_SMF.1	Specification of Management Functions
	FMT_SMR.2	Restrictions on Security Roles
Protection of the TSF	FPT_SKP_EXT.1	Extended: Protection of TSF Data (for reading of all symmetric keys)
	FPT_APW_EXT.1	Extended: Protection of Administrator Passwords
	FPT_STM.1	Reliable Time Stamps
	FPT_TST_EXT.1	TSF Testing (Extended)
	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

Class: Security Audit (FAU)

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 shutdown of the audit functions;

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

c) All administrator actions comprising:

- *Administrative login and logout (name of user account shall be logged if individual user accounts are required for administrators).*
- *Security related 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).*
- *Starting and stopping services (if applicable)*
- [*no other actions*];

d) [*Specifically defined auditable events listed in Table 12*.].

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 12*.

Table 12 Auditable Events

SFR	Auditable Event	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(1)	None.	None.
FCS_COP.1(2)	None.	None.

SFR	Auditable Event	Additional Audit Record Contents
FCS_COP.1(3)	None.	None.
FCS_COP.1(4)	None.	None.
FCS_HTTPS_EXT.1	Failure to establish an HTTPS session.	Reason for failure.
FCS_TLSS_EXT.1	Failure to establish an TLS session	Reason for failure.
FCS_TLSC_EXT.1	Failure to establish an TLS session	Reason for failure.
FCS_SSHS_EXT.1	Failure to establish an SSH session	Reason for failure
FCS_RBG_EXT.1	None.	None.
FIA_PMG_EXT.1	None.	None.
FIA_UIA_EXT.1(1) FIA_UIA_EXT.1(2)	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 identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UAU.7	None.	None.
FIA_X509_EXT.1	Unsuccessful attempt to validate a certificate	Reason for failure
FIA_X509_EXT.2	None.	None.
FIA_X509_EXT.3	None.	None.
FMT_MOF.1(1)/Audit	Modification of the behaviour of the transmission of audit data to an external IT entity.	None.
FMT_MOF.1(1)/Trusted Update	Any attempt to initiate a manual update	None.
FMT_MTD.1	All management activities of TSF data	None.
FMT_MTD.1/AdminAct	Modification, deletion, generation/import of cryptographic keys.	None.
FMT_SMF.1	None.	None.
FMT_SMR.2	None.	None.
FPT_SKP_EXT.1	None.	None.
FPT_APW_EXT.1	None.	None.

SFR	Auditable Event	Additional Audit Record Contents
FPT_STM.1	Changes to the time.	The old and new values for the time. Origin of the attempt to change time for success and failure (e.g., IP address).
FPT_TST_EXT.1	None.	None.
FPT_TUD_EXT.1	Initiation of update. result of the update attempt (success or failure)	No additional information.
FTA_SSL_EXT.1	Any attempts at unlocking of an interactive session.	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	Initiation of the trusted path. Termination of the trusted path. Failures of the trusted path functions.	Identification of the claimed user identity.

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.

FAU_STG.1 Protected Audit Trail Storage

FAU_STG.1.1 The TSF shall protect the stored audit records in the audit trail from unauthorised deletion.

FAU_STG.1.2 The TSF shall be able to prevent unauthorised modifications to the stored audit records in the audit trail.

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.

FAU_STG_EXT.1.3 The TSF shall [*rotate the audit log file*] when the local storage space for audit data is full.

Class: Cryptographic Support (FCS)

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: [

- **RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3;**
- **ECC schemes using "NIST curves" [P-256, P-384, P-521] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4**
- **FFC schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.1**

] ~~and specified cryptographic key sizes [assignment: cryptographic key sizes] that meet the following: [assignment: list of standards]~~.

FCS_CKM.2	Cryptographic Key Establishment

FCS_CKM.2.1 The TSF shall **perform** cryptographic **key establishment** in accordance with a specified cryptographic key **establishment** method: [

- **RSA-based key establishment schemes that meets the following: NIST Special Publication 800-56B Revision 1, "Recommendation for Pair-**

Wise Key Establishment Schemes Using Integer Factorization Cryptography";

- **Elliptic curve-based key establishment schemes that meets the following: NIST Special Publication 800-56A Revision 2, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography";**

- **Finite field -based key establishment schemes that meets the following: NIST Special Publication 800-56A Revision 2, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography"**

] ~~that meets the following: [assignment: list of standards].~~

FCS_CKM.4	Cryptographic Key Destruction

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method

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

that meets the following: *No Standard.*

FCS_COP.1(1) – Cryptographic Operation (AES Data Encryption/Decryption)

FCS_COP.1.1(1) The TSF shall perform *encryption/decryption* in accordance with a specified cryptographic algorithm *AES used in* [CBC, GCM] mode and cryptographic key sizes [128 bits, 256 bits] that meet the following: *AES as specified in ISO 18033-3*, [CBC as specified in ISO 10116, GCM as specified in ISO 19772].

FCS_COP.1 (2) – Cryptographic Operation (Signature Generation and Verification)

FCS_COP.1.1(2) The TSF shall perform *cryptographic signature services* **(***generation and verification*) in accordance with a specified cryptographic algorithm
[

- RSA Digital Signature Algorithm and cryptographic key sizes (**modulus**) [*2048 bits*],

- Elliptic Curve Digital Signature Algorithm and cryptographic key sizes [*256, 384, 521 bits*]

]

that meet the following:
[

- For RSA schemes: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSA-PKCS1v1_5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3,

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

].

FCS_COP.1(3) – Cryptographic Operation (Hash Algorithm)

FCS_COP.1.1(3) The TSF shall perform *cryptographic hashing services* in accordance with a specified cryptographic algorithm [SHA-1, SHA-256, SHA-384] ~~and cryptographic key sizes [assignment: cryptographic key sizes]~~ that meet the following: ISO/IEC 10118-3:2004.

FCS_COP.1(4) – Cryptographic Operation (Keyed Hash Algorithm)

FCS_COP.1.1(4) The TSF shall perform *keyed-hash message authentication* in accordance with a specified cryptographic algorithm [HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512] and cryptographic key sizes [*160, 256, 384, 512*] **and message digest sizes** [**160, 256, 384, 512**] **bits** that meet the following: ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2".

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

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 The TSF shall establish the connection only if [the peer initiates handshake].

FCS_TLSS_EXT.1 TLS Server Protocol

FCS_TLSS_EXT.1.1 The TSF shall implement [TLS 1.2 (RFC 5246)] supporting the following ciphersuites:

[

- TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289

].

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 [generate EC Diffie-Hellman parameters over NIST curves [secp256r1] and no other curves; generate Diffie-Hellman parameters of [2048 bits]].

FCS_TLSC_EXT.1 TLS Client Protocol

FCS_TLSC_EXT.1.1 The TSF shall implement [TLS 1.2 (RFC 5246)] supporting the following ciphersuites:

[

- TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492

- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289

].

FCS_TLSC_EXT.1.2 The TSF shall verify that the presented identifier matches the reference identifier according to RFC 6125.

FCS_TLSC_EXT.1.3 The TSF shall only establish a trusted channel if the peer certificate is valid.

FCS_TLSC_EXT.1.4 The TSF shall present the Supported Elliptic Curves Extension in the Client Hello with the following NIST curves: [secp256r1, secp384r1, secp521r1] and no other curves.

FCS_SSHS_EXT.1 SSH Server Protocol

FCS_SSHS_EXT.1.1 The TSF shall implement the SSH protocol that complies with RFCs 4251, 4252, 4253, 4254, and [6668].

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

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

FCS_SSHS_EXT.1.4 The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms and rejects all other encryption algorithms: [aes128-cbc, aes256-cbc, AEAD_AES_128_GCM, AEAD_AES_256_GCM].

FCS_SSHS_EXT.1.5 The TSF shall ensure that the SSH transport implementation uses [ssh-rsa, ecdsa-sha2-nistp256] and [ecdsa-sha2-nistp384, ecdsa-sha2-nistp521] as its public key algorithm(s) and rejects all other public key algorithms.

FCS_SSHS_EXT.1.6 The TSF that the SSH transport implementation uses [hmac-sha1 hmac-sha2-256, hmac-sha2-512] and [AEAD_AES_128_GCM, AEAD_AES_256_GCM] as its MAC algorithm(s) and rejects all other MAC algorithm(s).

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

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 no more than onegigabyte of transmitted data. After either of the thresholds are reached a rekey needs to be performed.

Class: Identification and Authentication (FIA)

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 settable by the Security Administrator, and support passwords of 15 characters or greater;*

FIA_UIA_EXT.1(1) User Identification and Authentication

FIA_UIA_EXT.1.1(1) The **HTTPS Web GUI Interface of 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 a list of the names and versions of installed software*].

FIA_UIA_EXT.1.2(1) The TSF shall require each administrative user to be successfully identified and authenticated before allowing any other TSF-mediated action on behalf of that administrative user.

Application Note: *This iteration of FIA_UIA_EXT.1 applies to the HTTPS Web GUI*

FIA_UIA_EXT.1(2) User Identification and Authentication

FIA_UIA_EXT.1.1(2) The **CLI Interface of 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(2) The TSF shall require each administrative user to be successfully identified and authenticated before allowing any other TSF-mediated action on behalf of that administrative user.

Application Note: *This iteration of FIA_UIA_EXT.1 applies to the CLI*

FIA_UAU_EXT.2 Password-based Authentication Mechanism

FIA_UAU_EXT.2.1 The TSF shall provide a local password-based authentication mechanism, and [none] to perform administrative user authentication.

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.

FIA_X509_EXT.1 X.509 Certificate Validation

FIA_X509_EXT.1.1 The TSF shall validate certificates in accordance with the following rules:

- RFC 5280 certificate validation and certificate path validation.
- The certificate path must terminate with a trusted CA certificate.
- The TSF shall validate 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 shall validate the revocation status of the certificate using [the Online Certificate Status Protocol (OCSP) as specified in RFC 2560].
- 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 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.

FIA_X509_EXT.2 X.509 Certificate Authentication

FIA_X509_EXT.2.1 The TSF shall use X.509v3 certificates as defined by RFC 5280 to support authentication for [TLS], and [no additional uses].

FIA_X509_EXT.2.2 When the TSF cannot establish a connection to determine the validity of a certificate, the TSF shall [accept the certificate].

FIA_X509_EXT.3 X.509 Certificate Requests

FIA_X509_EXT.3.1 The TSF shall generate a Certificate Request Message as specified by RFC 2986 and be able to provide the following information in the request: public key and [device-specific information, Common Name, Organization, Organizational Unit, Country].

FIA_X509_EXT.3.2 The TSF shall validate the chain of certificates from the Root CA upon receiving the CA Certificate Response.

Class: Security Management (FMT)

FMT_MOF.1(1)/TrustedUpdate - Management of security functions behavior

FMT_MOF.1.1/TrustedUpdate The TSF shall restrict the ability to enable the functions *to perform manual update* to *Security Administrators*.

FMT_MOF.1(1)/Audit - Management of security functions behavior

FMT_MOF.1.1(1)/Audit The TSF shall restrict the ability to determine the behaviour of, modify the behaviour of the functions *transmission of audit data to an external IT entity* to *Security Administrators*.

FMT_MOF.1(2)/Audit - Management of security functions behavior

FMT_MOF.1.1(2)/Audit The TSF shall restrict the ability to modify the behaviour of the functions *handling of audit data* to *Security Administrators.*

FMT_MOF.1(1)/AdminAct - Management of security functions behavior

FMT_MOF.1.1(1)/AdminAct The TSF shall restrict the ability to modify the behaviour of the functions *TOE Security Functions* to *Security Administrators.*

FMT_MOF.1(2)/AdminAct - Management of security functions behavior

FMT_MOF.1.1(2)/AdminAct The TSF shall restrict the ability to enable, disable the functions *services* to *Security Administrators*.

FMT_MTD.1 Management of TSF Data

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

FMT_MTD.1/AdminAct Management of TSF Data

FMT_MTD.1.1/AdminAct The TSF shall restrict the ability to *modify, delete, generate/import the cryptographic keys to Security Administrators*.

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 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 audit behavior;
- Ability to configure the cryptographic functionality]

FMT_SMR.2	Restrictions on Security Roles

FMT_SMR.2.1 The TSF shall maintain the roles:

- *Security Administrator*.

FMT_SMR.2.2 The TSF shall be able to associate users with roles.

FMT_SMR.2.3 The TSF shall ensure that the conditions

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

are satisfied.

Class: Protection of the TSF (FPT)

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.

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.

FPT_STM.1	Reliable Time Stamps

FPT_STM.1.1 The TSF shall be able to provide reliable time stamps.

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: [*FIPS 140-2 standard power-up self-tests and software/firmware integrity test*].

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 [no other TOE firmware/software version].

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 a means to authenticate firmware/software updates to the TOE using a [digital signature mechanism] prior to installing those updates.

Class: TOE Access (FTA)

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.

FTA_SSL.3 TSF-initiated Termination

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

FTA_SSL.4 User-initiated Termination

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

FTA_TAB.1 Default TOE Access Banners

FTA_TAB.1.1 Refinement: 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.

Class: Trusted Path/Channels (FTP)

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** 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 *[Syslog server over TLS].*

FTP_TRP.1	Trusted Path

FTP_TRP.1.1 The TSF shall **be capable of using [HTTPS, SSH] to** provide a communication path between itself and **authorized remote administrators** that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from *disclosure and provides detection of modification of the channel data.*

FTP_TRP.1.2 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**.

5.3 TOE SFR Dependencies Rationale for SFRs Found in NDcPP

The NDcPPv1.0 contains all the requirements claimed in this Security Target. As such the dependencies are not applicable since the cPP itself has been approved.

5.4 Security Assurance Requirements

SAR Requirements

The TOE assurance requirements for this ST are taken directly from the NDcPP which are derived from [CC_PART3]. The assurance requirements are summarized in the table below.

Table 13: Assurance Measures

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

Security Assurance Requirements Rationale

The Security Assurance Requirements (SARs) in this Security Target represent the SARs identified in the [NDcPP]. As such, the [NDcPP] SAR rationale is deemed acceptable since the cPP itself has been validated.

5.5 Assurance Measures

This section identifies the Assurance Measures applied by Cisco to satisfy the assurance requirements. The table below lists the details.

Table 14: Assurance Measures

Component	How requirement will be met
ADV_FSP.1	No additional "functional specification" documentation was provided by Cisco to satisfy the Evaluation Activities specified in the SD.
AGD_OPE.1 AGD_PRE.1	Guidance must include a description of how the IT personnel verifies that the Operational Environment can fulfill its role for the security functionality. The documentation should be in an informal style and readable by the IT personnel. Guidance must be provided for every operational environment that the product supports as claimed in the ST. This guidance includes: • instructions to successfully install the TSF in that environment; and • instructions to manage the security of the TSF as a product and as a component of the larger operational environment; and • instructions to provide a protected administrative capability. Guidance pertaining to particular security functionality must also be provided. Cisco will provide the guidance documents with the ST.
ALC_CMC.1 ALC_CMS.1	Cisco will identify the TOE such that it can be distinguished from other products or versions from the Cisco and can be easily specified when being procured by an end user.
ATE_IND.1	Cisco will provide the TOE for testing.
AVA_VAN.1	Cisco will provide the TOE for Vulnerability Analysis.

6 TOE SUMMARY SPECIFICATION

6.1 TOE Security Functional Requirement Measures

This chapter identifies and describes how the Security Functional Requirements identified above are met by the TOE.

Table 15: How TOE SFRs Measures

<table>
<tr><th>TOE SFRs</th><th>How the SFR is Met</th></tr>
<tr><td>FAU_GEN.1</td><td>The TOE generates an audit record that is stored internally within the TOE whenever an audited event occurs. The types of events that cause audit records to be generated include, cryptography related events, events related to the enforcement of information flow policies, identification and authentication related events, and administrative events. Table 12 lists the audit events as required by the NDcPP. This includes audit events for optional and selected SFRs in the ST.

See below for a list of events audited by the TOE:

<table>
<tr><th>Auditable Event</th><th>Rationale</th></tr>
<tr><td>Success and failure of encrypted communications (SSH, TLS, HTTPS) and successful SSH rekey</td><td>Attempts of secure encrypted communications/connections (SSH, TLS, HTTPS). The communications include the remote administrator establishing a session and the TOE sending syslog data. The identity of the non-TOE entity is included in the log record.</td></tr>
<tr><td>All use of the user identification and authentication mechanism.</td><td>Events will be generated for attempted identification/ authentication (including whether it was successful or failed), and the username attempting to authenticate will be included in the log record, along with the origin or source of the attempt.</td></tr>
<tr><td>Unsuccessful attempt to validate a certificate</td><td>The reason for failure of certificate validation attempts is logged.</td></tr>
<tr><td>Changes to the time.</td><td>Changes to the time are logged, including old and new values for time, as well as origin of attempt</td></tr>
<tr><td>Initiation of an update to the TOE.</td><td>TOE updates and the result of the update attempts are logged as configuration changes.</td></tr>
<tr><td>Termination of a remote session.</td><td>Termination of a remote session (due to inactivity) is logged (as a terminated cryptographic path).</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>TOE SFRs</th><th>How the SFR is Met</th></tr>
<tr><td></td><td>
<table>
<tr><td>Termination of an interactive session.</td><td>Termination of an Interactive session (due to logging off) is logged (as the session ending).</td></tr>
<tr><td>Initiation, termination and failures in trusted channels.</td><td>Requests for encrypted session negotiation are logged (including whether successful or failed). Similarly, when an established cryptographic channel or path is terminated or fails a log record is generated.
Also the initiator and target of any failed attempts to establish a trusted channels are identified.</td></tr>
<tr><td>Initiation, termination and failures in trusted paths.</td><td>Requests for encrypted session negotiation are logged (including whether successful or failed). Similarly, when an established cryptographic channel or path is terminated or fails a log record is generated.
The records include the claimed user identity.</td></tr>
<tr><td>All management activities of TSF data (e.g. Modification of the behaviour of the transmission of audit data to an external IT entity; Any attempt to initiate a manual update; Modification, deletion, generation/import of cryptographic keys; resetting passwords; starting/stopping services).</td><td>The use of the security management functions are logged, along with the origin or source of the attempt.</td></tr>
</table>
Additionally, the startup and shutdown of the audit functionality is audited.
</td></tr>
<tr><td>FAU_GEN.2</td><td>The TOE ensures each auditable event is associated with the user that triggered the event and as a result they are traceable to a specific user. For example a human user, user identity, or related session ID would be included in the audit record. For an IT entity or device, the IP address, host name, or other configured identification is presented.</td></tr>
<tr><td>FAU_STG_EXT.1
FAU_STG.1
FMT_MOF.1(1)/Audit
FMT_MOF.1(2)/Audit</td><td>All TOE audit data is stored in local log files that is rotated when full.
The TOE transmits audit event data to a specified, external syslog server. The TOE uses TCP syslog over the TLS protected trusted channel to transmit audit data to an external syslog server.
TOE audit data is viewed by successfully authenticating to the Command Line Interface (CLI) as an admin-role. The TOE restricts handling and access to the transmission of audit data to an external IT entity to the Security Administrator. Only the Security Administrator may modify transmission behaviour.</td></tr>
</table>

TOE SFRs	How the SFR is Met
	When Log files for the following audit events: o Resetting Passwords for Web GUI Admin Account o Add/Modify/Delete a Web GUI Admin Account o Failure to establish a HTTPS Session o Login and Logout events to the Web GUI reaches its maximum size, which is 10 MB by default, the log file is rotated. The Security Administrator may modify the size threshold as instructed in the AGD. Log files from all other audit events note listed above are rotated as follows: If the size reaches 10MB within a maximum of 7 days it will be rotated. It will also rotated weekly (Sunday), regardless of size. This is not configurable.
FCS_CKM.1 FCS_CKM.2	The TOE provides cryptographic functions to establish TLS, HTTPS, and SSH sessions. For HTTPS, the key generation for asymmetric keys implements RSA with key size 2048 bits according to FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3 and ECDSA over a NIST curve of P-256 bits according to FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4. Key establishment for asymmetric keys implements FFC and ECC-based key establishment scheme as specified in NIST SP 800-56A Revision 2 "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography". For TLS Client, the key generation for asymmetric keys implements RSA with key size 2048 bits according to FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3 and ECDSA over a NIST curve of P-256, P-384, and P-521 bits according to FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4. Key establishment for asymmetric keys implements FFC and ECC-based key establishment scheme as specified in NIST SP 800-56A Revision 2 "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography". For SSH, the key generation for asymmetric keys implements ECDSA over a NIST curve of P-256, P-384, and P-521 bits according to FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4. Key establishment for asymmetric keys implements FFC and ECC-based key establishment scheme as specified in NIST SP 800-56A Revision 2 "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" and RSA-based key establishment schemes as specified in NIST SP 800-56B Revision 1 "Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography" with key sizes greater than 112 bit key strength.

TOE SFRs	How the SFR is Met
	Refer to Table 17 for identification of the relevant CAVP certificates.
FCS_CKM.4	The TOE meets all requirements specified in FIPS 140-2 for destruction of keys and Critical Security Parameters (CSPs). The TOE destroys keys and Critical Security Parameters (CSPs) in that none of the symmetric keys or private keys are stored in plaintext form. The session keys used for symmetric encryption, private keys, and CSPs used to generate keys, are zeroized immediately after use, or on system shutdown. See Table 17, below for more information.
FCS_COP.1(1)	The TOE provides symmetric encryption and decryption capabilities using AES in CBC mode (128, 256) as specified in ISO 18033-3), in CBC mode (as specified in ISO 10116) and GCM mode (as specified in ISO 19772) with key sizes of 128 bits and 256 bits. AES is implemented in the following protocols: HTTPS, TLS, SSH.
FCS_COP.1(2)	The TOE provides cryptographic signature services using RSA Digital Signature Algorithm with key size of 2048 as specified in FIPS PUB 186-4, "Digital Signature Standard." Refer to Table 17 for identification of the relevant CAVP certificate.
FCS_COP.1(3)	The TOE provides cryptographic hashing services using SHA-1, SHA-256 and SHA-384. cryptographic hashing services are used in the HTTPS and TLS protocols. Refer to Table 17 for identification of the relevant CAVP certificate.
FCS_COP.1(4)	The TOE provides keyed-hashing message authentication services using HMAC-SHA-1(key size – 160 bits, block size 512 bits), HMAC-SHA-256 (key size – 256 bits, block size 512 bits) HMAC-SHA-384 (key size – 384 bits, block size 1024 bits), and SHA-512 (key size -512 bits, block size 1024 bits) and meets the ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2" standard. Refer to Table 18 for identification of the relevant CAVP certificate.
FCS_RBG_EXT.1	The TOE implements a random bit generator (RBG) based on the AES-256 block cipher, in accordance with ISO/IEC 18031:2011. The appliance form factor TOE uses the Emulex Pilot III BMC chips. The RBG for the PI appliance is seeded with a hardware-based noise source that uses a ring oscillator jitter based architecture that provides 256 bits of minimum entropy. Refer to Table 18 for identification of the relevant CAVP certificate.
FCS_HTTPS_EXT.1	The TOE provides HTTPS, as specified in RFC 2818, to provide a secure interactive interface for remote administrative functions, and to support secure exchange of user authentication parameters during login. HTTPS uses TLS to securely establish the encrypted remote session.

TOE SFRs	How the SFR is Met
FCS_TLSS_EXT.1 FCS_TLSC_EXT.1	The TOE implements TLS 1.2 conformant to RFC 5246 and supports the following ciphersuites: **FCS_TLSS_EXT.1** TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268 TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268 TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246 TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246 TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492 TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492 TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 **FCS_TLSC_EXT.1** TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268 TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268 TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492 TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492 TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246 TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246 TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 All connections from clients requesting SSL2.0, SSL3.0, TLS1.0, and TLS 1.1 are denied. The TOE only supports standard extensions, methods, and characteristics. TLS is used for management purposes and to establish encrypted sessions with IT entities to send/receive audit data. The TOE verifies that the presented identifier matches the reference identifier according to RFC 6125. When the TOE acts as a TLS client to syslog audit servers, it obtains the RFC 6125 reference identifiers from the administrator configured FQDN. The TOE supports SAN extension of type DNS name with a wildcard only in left-most label. Certificate pinning is unsupported by the TOE. For TLS Client connections, the TOE presents the Supported Elliptic Curves Extension in the Client Hello with the secp256r1, secp384r1, and secp521r1 NIST curves by default. No configuration is required. For key establishment the following key agreement parameters apply: For ECDHE, the TOE can generate 256-bit, 384-bit, or 521-bit ECDHE ephemeral keys. For FFC, the TOE can generate 2048-bit ephemeral key. For TLS Server connections, the TOE supports only the secp256r1 NIST curve and 256-bit ECDHE ephemeral key agreement parameter for

TOE SFRs	How the SFR is Met
	server key exchange. For FFC, the TOE can generate 2048-bit ephemeral key.
FCS_SSHS_EXT.1	The TOE implements SSHv2 and complies with RFCs 4251, 4252, 4253, 4254 and 6668. The TOE supports public-key authentication with ssh-rsa and ecdsa public key algorithms and password-based authentication. The TOE implementation of SSHv2 supports the following encryption algorithms - AES-128-CBC, AES-256-CBC, AES-128-GCM, and AES-256-GCM to ensure confidentiality of the session. SSHv2 connections will be dropped if the TOE receives a packet larger than 262,144 bytes. Large packets are detected by the SSHv2 implementation, and dropped internal to the SSH process. The TOE's implementation of SSHv2 supports hashing algorithms hmac-sha1, hmac-sha2-256, and hmac-sha2-512, AEAD_AES_128_GCM and AEAD_AES_256_GCM. The TOE's implementation of SSHv2 supports the followed key exchange algorithms: diffie-hellman-group14-sha1, ecdh-sha2-nistp256, ecdh-sha2-nistp384, and ecdh-sha2-nistp521. SSH session keys are rekeyed within one hour thresholds and when transmitted data exceeds onegigabyte.
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 (that include: "!", "@", "#", "$", "%", "^", "&", "*", "(", and ")". Minimum password length is settable by the Authorized Administrator, and can be configured for minimum password lengths of 15 characters or greater.
FIA_UIA_EXT.1(1) FIA_UIA_EXT.1(2)	Prior to login the HTTPS Web GUI, the TOE displays a list of the names and versions of all installed software updates and Security Administrator-specified advisory notice and consent warning message. Prior to login to the CLI, the TOE displays a Security Administrator-specified advisory notice and consent warning message only. The TOE requires all users to be successfully identified and authenticated before allowing any TSF mediated administrative actions to be performed. Administrative access to the TOE is facilitated through the TOE's CLI. The TOE mediates all administrative actions through the CLI. Once a potential administrative user attempts to access the CLI of the TOE through a directly connected console, remotely through a SSHv2 connection, or remotely through a TLS connection to the Web GUI, the TOE prompts the user for a user name and password. Only after the administrative user presents the correct authentication credentials will

<table>
<tr><th>TOE SFRs</th><th>How the SFR is Met</th></tr>
<tr><td></td><td>access to the TOE administrative functionality be granted. No access is allowed to the administrative functionality of the TOE until an administrator is successfully identified and authenticated.</td></tr>
<tr><td>FIA_UAU_EXT.2</td><td>The process for authentication is the same for administrative access whether administration is occurring remotely via SSHv2 and HTTPS web-based interface or via a local connection at the CLI. 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. The TOE then either grants administrative access (if the combination of username and password or public-key is correct) or indicates that the login was unsuccessful. The TOE does not provide a reason for failure in the cases of a login failure.
If the login was successful, the web GUI will display the home page. The CLI will display a command prompt like the following: PIServer/admin#.
The table below summarizes the authentication mechanisms that are supported at each interface.
<table>
<tr><th>Interface</th><th>Authentication Mechanism</th></tr>
<tr><td>Web-Based (GUI)</td><td>• local password-based</td></tr>
<tr><td>Remote SSH (CLI)</td><td>• SSH public key
• local password-based</td></tr>
<tr><td>Local Console (CLI)</td><td>• local password-based</td></tr>
</table>
</td></tr>
<tr><td>FIA_UAU.7</td><td>When an administrator enters their password at the CLI or GUI, each administrative interface displays only ‘*’ (asterisk) characters so that the password is obscured, or the TOE provides no feedback in the password field, and the TOE does not echo any characters back to remote clients as the characters are entered.</td></tr>
<tr><td>FIA_X509_EXT.1</td><td>The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for TLS connections. OCSP revocation checking is performed when authenticating a certificate provided by the remote server during TLS establishment. The certificate path is validated by ensuring that all the CA certificates has the basicConstraints extension and the CA flag is set to TRUE and the certificate path must terminate with a trusted CA certificate. The extendedKeyUsage field is validated according to the rules listed in FIA_X509_EXT.1.1.</td></tr>
<tr><td>FIA_X509_EXT.2</td><td>The certificates themselves are digitally signed and therefore are protected from tampering. In addition, the TOE identification and authentication security functions protect an unauthorized user from gaining access to the TOE. If a certificate is modified in any way, the digital signature verification process would invalidate it.

The validity check for the certificates takes place at session establishment and at time of import.</td></tr>
</table>

TOE SFRs	How the SFR is Met
	If a connection cannot be established to determine the revocation status of a certificate, the TOE will accept the certificate.
FIA_X509_EXT.3	A Certificate Request Message can be generated as specified by RFC 2986 and provide the following information in the request – FQDN of the PI Server, OU, O, City, State, and Country . The TOE can validate the chain of certificates from the Root CA when the CA Certificate Response is received.
FMT_MOF.1(1)/ TrustedUpdate FMT_MTD.1 FMT_MOF.1(1)/ AdminAct FMT_MOF.1(2)/ AdminAct	The TOE restricts the ability to enable the functions to perform manual update to the Security Administrator. The TOE restricts access to the management functions to the Security Administrator. The TOE supports two levels of administrators, the CLI-admin (local console) and the web-based admin user. The same functionality is available on the TOE via the web-based interface and CLI, with the exception that only the CLI-admin can start and stop the PI Services application and reload (update) or shutdown the appliance via the CLI. None of the administrative functions of the product are available prior to administrator log-in.
FMT_SMF.1 FMT_MTD.1/ AdminAct	The TOE provides all the capabilities necessary to securely manage the TOE, the services provided by the TOE. The management functionality of the TOE is provided through the TOE CLI (local or SSH) or HTTPS web-based interface. The specific management capabilities available from the TOE are identified in the text of the SFR - FMT_SMF.1. The Security administrator has the ability to generate, delete and import/export cryptographic keys.
FMT_SMF.1	The TOE provides all the capabilities necessary to securely manage the TOE. The Security Administrators can connect to the TOE to perform management functions via a directly connected console cable or remotely over TLS/HTTPS or SSH and can perform specific management capabilities including, but not limited to: • Local and remote administration of the TOE and the services provided by the TOE; • Configure an advisory notice and consent warning message to be displayed at login prior to gaining access to administrative functions; • Define the length of time that an administrative session can remain inactive before the session is terminated, and can configure serial console and TLS with separate timeout limits; • Initiate updates of the TOE software, including certificate-based image integrity verification; • Configure the cryptographic functionality;

TOE SFRs	How the SFR is Met
	• Enable or disable logging to the local audit log, or to the local console, or to remote syslog servers, and to display the configuration and status of audit functions.
FMT_SMR.2	The provides role-based access control that ensures security by restricting access privileges. A standard set of privileges is paired with each predefined role. A user can be assigned to multiple roles, which provides them with privileges for each role to which they are assigned. An individual who manages or performs a specific type of administrative task using the web GUI interface is considered an admin (or administrator). Administrators are dependent upon the admin role assigned to them, which limits the network access or tasks they can perform (a role-based access approach.)
FPT_SKP_EXT.1	The TOE is designed specifically to not disclose any keys stored in the TOE. The TOE encrypts and stores all private keys in a secure directory that is not readily accessible to administrators. There is no administrative interface provided to directly access the keys. The TOE stores symmetric keys only in volatile memory.
FPT_APW_EXT.1	The TOE is designed specifically to not disclose any passwords stored in the TOE. The TOE stores passwords in a SHA-2 hash format that's not accessible even to the Security Administrator. 'Show' commands at each administrative interface displays only '*' (asterisk) characters obscuring the password.
FPT_STM.1	The TOE maintains a reliable source of date and time and provides the Security Administrator an administrative capability to set date and time values. Time is reliable as the clock function is reliant on the real-time clock (RTC) provided by the underlying hardware. The TOE relies upon date and time information for the following security functions: • Monitoring local and remote interactive administrative sessions for inactivity. • Validating X.509 certificates to determine if a certificate has expired. • Timestamps in audit records.
FPT_TST_EXT.1	The TOE runs a suite of self-tests during initial start-up to verify correct operation of cryptographic modules. These tests are sufficient to verify that the correct version of the TOE software is running as well as that the cryptographic operations are all performing as expected. These tests include: • AES Known Answer Test - For the encrypt test, a known key is used to encrypt a known plain text value resulting in an encrypted value. This encrypted value is compared to a known encrypted value to ensure that the encrypt operation is working correctly.

TOE SFRs	How the SFR is Met
	The decrypt test is just the opposite. In this test a known key is used to decrypt a known encrypted value. The resulting plaintext value is compared to a known plaintext value to ensure that the decrypt operation is working correctly. • HMAC Known Answer Test - For each of the hash values listed, the HMAC implementation is fed known plaintext data and a known key. These values are used to generate a MAC. This MAC is compared to a known MAC to verify that the HMAC and hash operations are operating correctly. • DRBG Known Answer Test - For this test, known seed values are provided to the DRBG implementation. The DRBG uses these values to generate random bits. These random bits are compared to known random bits to ensure that the DRBG is operating correctly. • SHA-1/256/512 Known Answer Test – For each of the values listed, the SHA implementation is fed known data and key. These values are used to generate a hash. This hash is compared to a known value to verify they match and the hash operations are operating correctly. • HMAC (HMAC-SHA-1/256/512) KATs - For each of the hash values listed, the HMAC implementation is fed known plaintext data and a known key. These values are used to generate a MAC. This MAC is compared to a known MAC to verify that the HMAC and hash operations are operating correctly. • Software Integrity Test (HMAC-SHA1). RSA Signature Known Answer Test (both signature/verification) - This test takes a known plaintext value and Private/Public key pair and used the public key to encrypt the data. This value is compared to a known encrypted value to verify that encrypt operation is working properly. The encrypted data is then decrypted using the private key. This value is compared to the original plaintext value to ensure the decrypt operation is working properly. If any of the self-tests fail, the administrative UI will not be accessible. For a limited time window the Security Administrator will be able to login to the local CLI console. After authenticating, a fatal error is displayed and is only allowed to press <Enter> to logout and no other actions can be performed. The error message that will be displayed Error: NCS Services have been disabled because FIPS integrity check has failed! Either reimage from installation media, or contact Cisco Technical Support for instructions on diagnosing the failure.
FPT_TUD_EXT.1	The Security Administrator can query the software version running on the TOE and can initiate updates (new software images). When an update is made available by Cisco, the Security Administrator can manually obtain the update from the Cisco website and install it.

TOE SFRs	How the SFR is Met
	Updates are downloaded from the Cisco Care Online (CCO) at software.cisco.com. During installation of a TOE update, a digital signature verification check will automatically be performed to ensure it has not been modified since distribution. The authorized source for the digitally signed updates is "Cisco Systems, Inc.". If the digital signature is verified the update will be applied. When the trusted update has completed, the Security Administrator should log in and check the version on the Software Update page. If the digital signature fails to verity, or if update does not complete, an error will appear. Contact Cisco Technical Support for assistance.
FTA_SSL_EXT.1 FTA_SSL.3	The Security Administrator can configure maximum inactivity times individually for both local and remote administrative sessions. When a session is inactive (i.e., no session input) 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
FTA_SSL.4	The Security Administrator is able to exit out of both local and remote administrative sessions. The Security Administrator can logout of the web GUI by clicking *Logout* in the top-right corner of the page. Using SSH, the Security Administrator can logout by entering *exit*.
FTA_TAB.1	The Security Administrator may define a custom login banner that will be displayed to users of the TOE who connect locally to the serial console or remotely to the web GUI and SSHv2 sessions.
FTP_ITC.1	The TOE uses TLS to protect communications between itself and a remote syslog server. The TOE initiates trusted channel communication between itself and a remote syslog. The TOE acts as a TLS client and only TLS 1.2 is allowed. Refer to FCS_TLSC_EXT.1. The TOE identifies the remote syslog server using a FQDN reference identifier configured by the Security Administrator. The authenticates the device with X.509v3 certificates.
FTP_TRP.1	All remote administrative communications take place over a secure encrypted SSHv2 (CLI) session or HTTPS/TLS (web-based GUI) session. Both SSHv2 and HTTPS sessions are protected using AES encryption. The remote users are able to initiate both TLS and SSHv2 communications with the TOE and is required to successfully authenticate and be authorized for the role of Security Administrator before any remote administrative actions may be performed.

7 SUPPLEMENTAL TOE SUMMARY SPECIFICATION INFORMATION

7.1 Key Zeroization

The following table describes the key zeroization referenced by FCS_CKM.4 provided by the TOE.

Table 16: TOE Key Zeroization

Name	Description	Zeroization
TLS Private Key	The private key is used for HTTPS and TLS client connections. The private key is stored on the local filesystem and in DRAM.	Generation of a new certificate. Overwritten with: 0x00
TLS Master Key	Used for generating sessions encryption keys, MAC secrets. Stored in DRAM.	Automatically when the session is terminated. Overwritten with: 0x00
TLS Session Keys	Client write key: The key used to encrypt data written by the client and sent to the TLS server. Stored in DRAM. Client write MAC secret: The secret data used to authenticate data written by the client. Stored in DRAM.	Automatically when the session is terminated. Overwritten with: 0x00
SSH Private Key	Once the function has completed the operations requiring the RSA key object, the module over writes the entire object (no matter its contents) via API call. This overwrites the key with all 0's. The SSH server host private key is stored on the local filesystem and in DRAM.	Generation of a new key Overwritten with: 0x00
SSH Session Key	The results zeroized by overwriting the values with 0x00. This is done when a session is ended. This key is stored in DRAM.	Automatically when the SSH session is terminated. Overwritten with: 0x00

7.2 CAVP Certificates

The TOE processors are an Intel(R) Xeon(R) CPU E5-2650 v3 @ 2.30GHz
The TOE executes upon the Linux 2.6 kernel.

See table 18 below for CAVP certificates.

Table 17: CAVP Certificates

SFR	Algorithm	CAVP Certificate Number
FCS_CKM.1 – Cryptographic Key Generation – RSA	RSA	1743
FCS_CKM.1 – Cryptographic Key Generation – ECC	ECDSA	678
FCS_CKM.1 – Cryptographic Key Generation – FFC	DSA	961
FCS_CKM.2 – Cryptographic Key Establishment – RSA (Also see RSA #1743, SHS #2817, DRBG #817 in this table)	RSA	N/A
FCS_CKM.2 – Cryptographic Key Establishment – ECC FCS_CKM.2 – Cryptographic Key Establishment – FFC	CVL	1112
FCS_COP.1(1) - Cryptographic Operation – AES Encryption/Decryption	AES	3404
FCS_COP.1 (2) – Cryptographic Operation (Signature Generation and Verification) – RSA	RSA	1743
FCS_COP.1 (2) – Cryptographic Operation (Signature Generation and Verification) – ECDSA	ECDSA	678
FCS_COP.1(3) – Cryptographic Operation - Hashing Algorithms	SHS	2817
FCS_COP.1(4) – Cryptographic Operation - Keyed Hash	HMAC	2172
FCS_RBG – Random Bit Generation	DRBG	817

8 ANNEX A: REFERENCES

The following documentation was used to prepare this ST:

Table 18: References

Identifier	Description
[CC_PART1]	Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and general model, dated September 2012, version 3.1, Revision 4, CCMB-2012-09-001
[CC_PART2]	Common Criteria for Information Technology Security Evaluation – Part 2: Security functional components, dated September 2012, version 3.1, Revision 4, CCMB-2012-09-002
[CC_PART3]	Common Criteria for Information Technology Security Evaluation – Part 3: Security assurance components, dated September 2012, version 3.1, Revision 4, CCMB-2012-09-003
[CEM]	Common Methodology for Information Technology Security Evaluation – Evaluation Methodology, dated September 2012, version 3.1, Revision 4, CCMB-2012-09-004
[NDcPP]	collaborative Protection Profile for Network Devices, version 1.0 February 27, 2015
[NDSP]	Supporting Document – Evaluation Activities for Network Device cPP, version 1.0, February 27, 2015 CCDB-2015-01-001