Layer 7 API Gateway v11.1

Security Target

Version 1.1

July 2026

Document prepared by

www.lightshipsec.com

Document History

Version	Date	Author	Description
1.0	16 Jul 2026	G. NICKEL	Release for certification
1.1	24 Jul 2026	M Baldock	AGD version update

Table of Contents

List of Tables

1 Introduction

1.1 Overview

1 This Security Target (ST) defines the Layer 7 API Gateway v11.1 Target of Evaluation (TOE) for the purposes of Common Criteria (CC) evaluation.

2 The Layer7 API Gateway is a high-performance gateway that connects data and applications across cloud, container or on-premises environments. The Layer7 API Gateway is a service gateway that controls how web services are exposed to and accessed by external client applications.

1.2 Identification

Table 1: Evaluation identifiers

Target of Evaluation	Layer 7 API Gateway v11.1 Build: 11.1.3-24843
Security Target	Layer 7 API Gateway v11.1 Security Target, v1.1

1.3 Conformance Claims

3 This ST supports the following conformance claims:

a) CC version 3.1 revision 5

b) CC Part 2 extended

c) CC Part 3 conformant

d) collaborative Protection Profile for Network Devices, v3.0e (referenced within as NDcPP)

e) Functional Package for SSH, v1.0 (reference within as PKG_SSH) conformant

f) NIAP Technical Decisions per Table 2

Table 2: NIAP Technical Decisions

TD #	Name	Source	Applicability Rational
TD0682	Addressing Ambiguity in FCS_SSHS_EXT.1 Tests	PKG_SSH	Applicable.
TD0695	Choice of 128 or 256 bit size in AES-CTR in SSH Functional Package.	PKG_SSH	Applicable.
TD0732	FCS_SSHS_EXT.1.3 Test 2 Update	PKG_SSH	Applicable.
TD0777	Clarification to Selections for Auditable Events for FCS_SSH_EXT.1	PKG_SSH	Applicable.

TD #	Name	Source	Applicability Rational
TD0909	Updates to FCS_SSH_EXT.1.1 App Note in SSH FP 1.0	PKG_SSH	Applicable.
TD0967	Allowance of Kex-strict in PKG_SSH_V1.0	PKG_SSH	Applicable.
TD1015	Addition of ECD to PKG_SSH_V1.0	PKG_SSH	Applicable.
TD1023	Clarifications to FCS_SSH_EXT.1.5 When aes256-gcm@openssh.com Is Selected	PKG_SSH	Applicable.
TD0836	NIT Technical Decision: Redundant Requirements in FPT_TST_EXT.1	NDcPP	Applicable.
TD0868	NIT Technical Decision: Clarification of time frames in FCS_IPSEC_EXT.1.7 and FCS_IPSEC_EXT.1.8	NDcPP	Not Applicable. IPSEC not claimed.
TD0879	NIT Decision: Correction of Chapter Headings in CPP_ND_V3.0E	NDcPP	Applicable.
TD0880	NIT Decision: Removal of Duplicate Selection in FMT_SMF.1.1	NDcPP	Applicable.
TD0886	Clarification to FAU_STG_EXT.1 Test 6	NDcPP	Applicable.
TD0899	NIT Technical Decision: Correction of Renegotiation Test for TLS 1.2	NDcPP	Applicable
TD0900	NIT Technical Decision: Clarification to Local Administrator Access in FIA_UIA_EXT.1.3	NDcPP	Applicable.
TD0921	NIT Technical Decision: Addition of FIPS PUB 186-5 and Correction of Assignment	NDcPP	Applicable.
TD0923	NIT Technical Decision: Auditable event for FAU_STG_EXT.1 in FAU_GEN.1.2	NDcPP	Applicable.
TD0973	NIT Technical Decision: FCS_(D)TLSS_EXT.1.3 Test 2 DHE Ciphersuite Conditionality	NDcPP	Applicable.

TD #	Name	Source	Applicability Rational
TD0990	NIT Technical Decision: CTR_DRBG in FCS_RBG_EXT.1.2	NDcPP	Applicable.
TD1033	Sunset Dates for NDcPP Configurations	NDcPP	Applicable.

1.4 Terminology

Table 3: Terminology

Term	Definition
CC	Common Criteria
EAL	Evaluation Assurance Level
NDcPP	collaborative Protection Profile for Network Devices
PP	Protection Profile
TOE	Target of Evaluation
TSF	TOE Security Functionality

2 TOE Description

2.1 Type

4 The Layer7 API Gateway (TOE) is a high-performance gateway that connects data and applications across cloud, container or on-premises environments. The TOE is a service gateway that controls how web services are exposed to and accessed by external client applications.

5 The TOE is a Case 1 virtual network device.

2.2 Usage

2.2.1 Deployment

6 The TOE (enclosed in red) is a virtual appliance deployed in a network that provides access control to a corporate network. The TOE provides access control to web services that are exposed to and accessed by external client applications.

7 Figure 1 depicts an example deployment of the TOE.

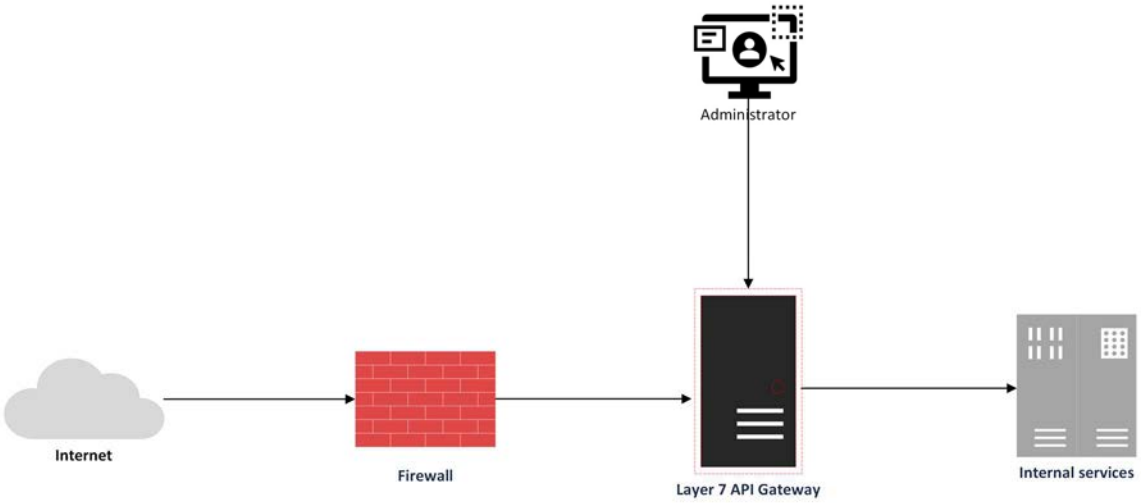

Figure 1: Example TOE deployment

2.2.2 Interfaces

8 The TOE management interfaces are shown in Figure 2.

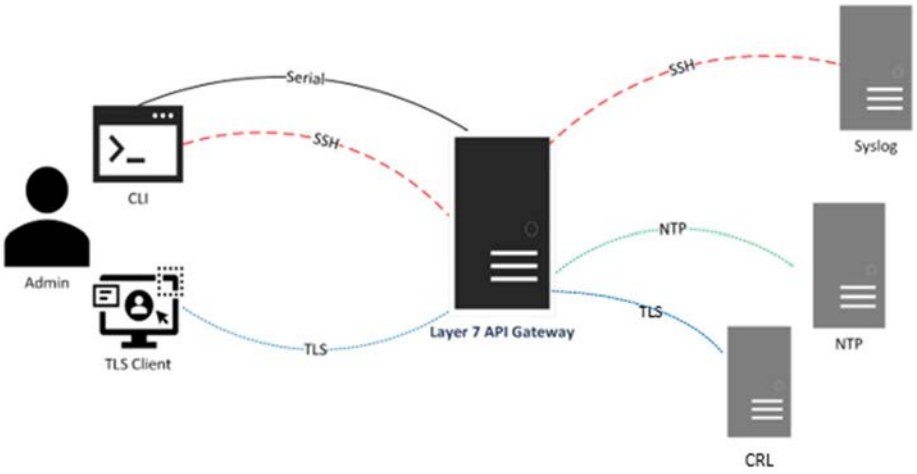

Figure 2: TOE interfaces

9 The TOE interfaces are as follows:

a) **CLI.** Command line management interface via virtual console or remote SSH.

b) **TLS Client.** Policy Manager, management interface via TLS.

c) **Logs.** Transmission of logs to a syslog server via SSH**.**

d) **NTP.** The TOE synchronizes time via NTP.

e) **CRL Responder.** The TOE uses a CRL for certificate revocation checking.

2.3 Security Functions / Logical Scope

10 The TOE provides the following security functions:

a) **Trusted Path/Channels.** The TOE protects the integrity and confidentiality of communications as noted in section 2.2.2 above, and using cryptographic algorithms as described in Table 4.

b) **Security Management.** The TOE enables secure management of its security functions, including:

i) Administrator authentication with passwords and public keys

ii) Configurable password policies

iii) Role Based Access Control

iv) Access banners

v) Management of critical security functions and data

vi) Protection of cryptographic keys and passwords

vii) Trusted updates via digital signature validation

c) **Protection of the TSF.** The TOE performs a suite of self-tests to ensure the correct operation and enforcement of its security functions. The TOE performs diagnostic self-tests and cryptographic module self-tests during start-up and generates audit records to record a failure. Self-tests comply with the FIPS 140-2 requirements for self-testing.

d) **Identification and Authentication.** The TOE ensures that all users must be authenticated before accessing its functions and data. The TOE uses X.509 certificates to support authentication for TLS. Certificate revocation checking is performed using a CRL. The TOE uses public keys for authentication for SSH.

e) **TOE Access.** TOE can be accessed directly via serial connection or remotely via TLS and SSH connections. When a user account has sequentially failed authentication the configured number of times, the account will be locked for a Security Administrator defined time period.

f) **Security Audit.** The TOE generates audit records of user and administrator actions. The TOE includes the user identity in audit events resulting from actions of identified users. The Security Administrator can configure the TOE to send logs in real-time to a syslog server via SSH.

g) **Cryptographic Support** The TOE implements a cryptographic module. The cryptographic module has the ability to generate and destroy cryptographic keys. Relevant Cryptographic Algorithm Validation Program (CAVP) certificates are shown in Table 4.

Table 4: CAVP Certificates

Algorithm Capability	Certificate
AES-CBC-128 AES-CBC-256	A8755 A7490
AES-GCM-128 AES-GCM-256	
KAS-FFC	
AES-CTR-128 AES-CTR-256	
ECDSA Key Gen (186-4) ECDSA Sig Gen (186-4) ECDSA Sig Ver (186-4)	
RSA Key Gen (186-4) RSA Sig Gen (186-4) RSA Sig Ver (186-4)	
SHA-1, SHA-256, SHA-384, SHA-512	
HMAC-SHA-256, HMAC-SHA-512	
KAS-ECC	
Counter DRBG HMAC DRBG	

2.4 Physical Scope

11 The TOE boundary includes the Gateway 11.1.00-17707 VMware Debian12 OVA upgraded with a number of L7P patches, that runs inside a virtual machine. The TOE is downloaded from the Broadcom portal.

Type	Model	CPU	Software
Virtual Machine	ESXi 8.0	Intel(R) Xeon(R) Gold 6240 CPU @ 2.60GHz	11.1.3-24843

2.4.1 Guidance Documents

12 The TOE includes the following guidance documents (PDF):

a) Broadcom Layer 7 API Gateway v11.1 Common Criteria Guide, v1.1

13 Other supplemental guidance includes:

a) Common Criteria Installation and Configuration Guide for Gateway 11.1.x, May 11, 2026

b) Guidance documents from Broadcom's web portal. https://techdocs.broadcom.com/us/en/ca-enterprise-software/layer7-api-management/api-gateway/11-1.html

2.4.2 Non-TOE Components

14 The TOE operates with the following components in the environment:

a) **Policy Manager:** The Administrator accesses the TLS management interface via a Vendor provided GUI application. This application performs X.509 certificate path validation similar to a Web Browser.

b) **Audit Server.** The TOE sends audit events to a syslog server.

c) **NTP Server.** The TOE synchronizes time via NTP.

d) **VMware hypervisors (ESX, ESXi, vSphere).** The TOE operates on VMware ESXi 8.0.

e) **CRL Responder.** The TOE uses a CRL for certificate revocation checking.

2.4.3 Functions not included in the TOE Evaluation

15 The functions evaluated are limited to those identified at section 2.3. The following functions have not been assessed as part of this evaluation:

a) XML firewall and policy enforcement features.

b) vSphere High Availability feature

c) Identity and Access Management feature

d) Policy Manager Java Client.

3 Security Problem Definition

16 The Security Problem Definition is reproduced from section 4 of the NDcPP.

3.1 Threats

Table 5: Threats

Identifier	Description
T.UNAUTHORIZED_ ADMINISTRATOR_ ACCESS	Threat agents may attempt to gain Administrator access to the Network Device by nefarious means such as masquerading as an Administrator to the device, masquerading as the device to an Administrator, replaying an administrative session (in its entirety, or selected portions), or performing man-in-the-middle attacks, which would provide access to the administrative session, or sessions between Network Devices. Successfully gaining Administrator access allows malicious actions that compromise the security functionality of the device and the network on which it resides.
T.WEAK_ CRYPTOGRAPHY	Threat agents may exploit weak cryptographic algorithms or perform a cryptographic exhaust against the key space. Poorly chosen encryption algorithms, modes, and key sizes will allow attackers to compromise the algorithms, or brute force exhaust the key space and give them unauthorized access allowing them to read, manipulate and/or control the traffic with minimal effort.
T.UNTRUSTED_ COMMUNICATION_ CHANNELS	Threat agents may attempt to target Network Devices that do not use standardized secure tunnelling protocols to protect the critical network traffic. Attackers may take advantage of poorly designed protocols or poor key management to successfully perform man-in-the-middle attacks, replay attacks, etc. Successful attacks will result in loss of confidentiality and integrity of the critical network traffic, and potentially could lead to a compromise of the Network Device itself.
T.WEAK_ AUTHENTICATION_ ENDPOINTS	Threat agents may take advantage of secure protocols that use weak methods to authenticate the endpoints, e.g. a shared password that is guessable or transported as plaintext. The consequences are the same as a poorly designed protocol, the attacker could masquerade as the Administrator or another device, and the attacker could insert themselves into the network stream and perform a man-in-the-middle attack. The result is the critical network traffic is exposed and there could be a loss of confidentiality and integrity, and potentially the Network Device itself could be compromised.
T.UPDATE_ COMPROMISE	Threat agents may attempt to provide a compromised update of the software or firmware which undermines the security functionality of the device. Non-validated updates or updates validated using non-secure or weak cryptography leave the update firmware vulnerable to surreptitious alteration.
T.UNDETECTED_ ACTIVITY	Threat agents may attempt to access, change, and/or modify the security functionality of the Network Device without Administrator awareness. This could result in the attacker finding an avenue (e.g., misconfiguration, flaw in the product) to compromise the device and

Identifier	Description
	the Administrator would have no knowledge that the device has been compromised.
T.SECURITY_ FUNCTIONALITY_ COMPROMISE	Threat agents may compromise credentials and device data enabling continued access to the Network Device and its critical data. The compromise of credentials includes replacing existing credentials with an attacker's credentials, modifying existing credentials, or obtaining the Administrator or device credentials for use by the attacker. Threat agents may also be able to take advantage of weak administrative passwords to gain privileged access to the device.
T.SECURITY_ FUNCTIONALITY_ FAILURE	An external, unauthorized entity could make use of failed or compromised security functionality and might therefore subsequently use or abuse security functions without prior authentication to access, change or modify device data, critical network traffic or security functionality of the device.

3.2 Assumptions

Table 6: Assumptions

Identifier	Description
A.PHYSICAL_ PROTECTION	The Network Device is assumed to be physically protected in its operational environment and not subject to physical attacks that compromise the security or interfere with the device's physical interconnections and correct operation. This protection is assumed to be sufficient to protect the device and the data it contains. As a result, the cPP does not include any requirements on physical tamper protection or other physical attack mitigations. The cPP does not expect the product to defend against physical access to the device that allows unauthorized entities to extract data, bypass other controls, or otherwise manipulate the device. For vNDs, this assumption applies to the physical platform on which the VM runs.
A.LIMITED_ FUNCTIONALITY	The device is assumed to provide networking functionality as its core function and not provide functionality/services that could be deemed as general purpose computing. For example, the device should not provide a computing platform for general purpose applications (unrelated to networking functionality). If a virtual TOE evaluated as a pND, following Case 2 vNDs as specified in Section 1.2, the VS is considered part of the TOE with only one vND instance for each physical hardware platform. The exception being where components of a distributed TOE run inside more than one virtual machine (VM) on a single VS. In Case 2 vND, no non-TOE guest VMs are allowed on the platform.

Identifier	Description
A.NO_THRU_ TRAFFIC_ PROTECTION	A standard/generic Network Device does not provide any assurance regarding the protection of traffic that traverses it. The intent is for the Network Device to protect data that originates on or is destined to the device itself, to include administrative data and audit data. Traffic that is traversing the Network Device, destined for another network entity, is not covered by the ND cPP. It is assumed that this protection will be covered by cPPs and PP-Modules for particular types of Network Devices (e.g., firewall).
A.TRUSTED_ ADMINISTRATOR	The Security Administrator(s) for the Network Device are assumed to be trusted and to act in the best interest of security for the organization. This includes appropriately trained, following policy, and adhering to guidance documentation. Administrators are trusted to ensure passwords/credentials have sufficient strength and entropy and to lack malicious intent when administering the device. The Network Device is not expected to be capable of defending against a malicious Administrator that actively works to bypass or compromise the security of the device. For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are expected to fully validate (e.g. offline verification) any CA certificate (root CA certificate or intermediate CA certificate) loaded into the TOE's trust store (aka 'root store', ' trusted CA Key Store', or similar) as a trust anchor prior to use (e.g. offline verification).
A.REGULAR_ UPDATES	The Network Device firmware and software is assumed to be updated by an Administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
A.ADMIN_ CREDENTIALS_ SECURE	The Administrator's credentials (private key) used to access the Network Device are protected by the platform on which they reside.
A.RESIDUAL_ INFORMATION	The Administrator must ensure that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment.
A.VS_TRUSTED_AD MINISTRATOR (applies to vNDs only)	The Security Administrators for the VS are assumed to be trusted and to act in the best interest of security for the organization. This includes not interfering with the correct operation of the device. The Network Device is not expected to be capable of defending against a malicious VS Administrator that actively works to bypass or compromise the security of the device.
A.VS_REGULAR_UP DATES (applies to vNDs only)	The VS software is assumed to be updated by the VS Administrator on a regular basis in response to the release of product updates due to known vulnerabilities.

Identifier	Description
A.VS_ISOLATION (applies to vNDs only)	For vNDs, it is assumed that the VS provides, and is configured to provide sufficient isolation between software running in VMs on the same physical platform. Furthermore, it is assumed that the VS adequately protects itself from software running inside VMs on the same physical platform.
A.VS_CORRECT_CONFIGURATION (applies to vNDs only)	For vNDs, it is assumed that the VS and VMs are correctly configured to support ND functionality implemented in VMs.

3.3 Organizational Security Policies

Table 7: Organizational Security Policies

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

4 Security Objectives

17 The security objectives are reproduced from section 5 of the NDcPP.

Table 8: Security Objectives for the Operational Environment

Identifier	Description
OE.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.
OE.NO_GENERAL_PURPOSE	There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE. Note: For vNDs the TOE includes only the contents of the its own VM, and does not include other VMs or the VS.
OE.NO_THRU_TRAFFIC_PROTECTION	The TOE does not provide any protection of traffic that traverses it. It is assumed that protection of this traffic will be covered by other security and assurance measures in the operational environment.
OE.TRUSTED_ADMIN	Security Administrators are trusted to follow and apply all guidance documentation in a trusted manner. For vNDs, this includes the VS Administrator responsible for configuring the VMs that implement ND functionality. For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are assumed to monitor the revocation status of all certificates in the TOE's trust store and to remove any certificate from the TOE's trust store in case such certificate can no longer be trusted.

Identifier	Description
OE.UPDATES	The TOE firmware and software is updated by an Administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
OE.ADMIN_ CREDENTIALS_ SECURE	The Administrator's credentials (private key) used to access the TOE must be protected on any other platform on which they reside.
OE.RESIDUAL_ INFORMATION	The Security Administrator ensures that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment. For vNDs, this applies when the physical platform on which the VM runs is removed from its operational environment.
OE.VM_CONFIGURATION (applies to vNDs only)	For vNDs, the Security Administrator ensures that the VS and VMs are configured to: • reduce the attack surface of VMs as much as possible while supporting ND functionality (e.g., remove unnecessary virtual hardware, turn off unused inter-VM communications mechanisms), and • correctly implement ND functionality (e.g., ensure virtual networking is properly configured to support network traffic, management channels, and audit reporting). The VS should be operated in a manner that reduces the likelihood that vND operations are adversely affected by virtualisation features such as cloning, save/restore, suspend/resume, and live migration. If possible, the VS should be configured to make use of features that leverage the VS's privileged position to provide additional security functionality. Such features could include malware detection through VM introspection, measured VM boot, or VM snapshot for forensic analysis.

5 Security Requirements

5.1 Conventions

18 This document uses the following font conventions to identify the operations defined by the CC:

a) **Assignment.** Indicated with italicized text.

b) **Refinement.** Indicated with bold text and strikethroughs.

c) **Selection.** Indicated with underlined text.

d) **Assignment within a Selection:** Indicated with italicized and underlined text.

e) **Iteration.** Indicated by adding a string starting with "/" (e.g. "FCS_COP.1/Hash").

19 **Note:** Operations performed within the Security Target are denoted within brackets []. Operations shown without brackets are reproduced from the NDcPP.

5.2 Extended Components Definition

20 The Extended Components are defined in Appendix C of the NDcPP.

Table 9: Extended Components

Requirement	Title	Source	Applicable TDs
FAU_STG_EXT.1	Protected Audit Event Storage	NDcPP	TD0886, TD0923
FCS_RBG_EXT.1	Random Bit Generation	NDcPP	TD0990
FCS_NTP_EXT.1	NTP Protocol	NDcPP	
FCS_SSH_EXT.1	SSH Protocol	PKG_SSH	TD0695, TD0777, TD0909, TD0967, TD1015, TD1023
FCS_SSHC_EXT.1	SSH Protocol – Client	PKG_SSH	TD1015
FCS_SSHS_EXT.1	SSH Protocol – Server	PKG_SSH	TD0682, TD0732, TD1015
FCS_TLSS_EXT.1	TLS Server Protocol	NDcPP	TD0899, TD0973
FIA_PMG_EXT.1	Password Management	NDcPP	
FIA_UIA_EXT.1	User Identification and Authentication	NDcPP	TD0900
FIA_X509_EXT.1/Rev	X.509 Certificate Validation	NDcPP	
FIA_X509_EXT.2	X.509 Certificate Authentication	NDcPP	
FIA_X509_EXT.3	X.509 Certificate Requests	NDcPP	

Requirement	Title	Source	Applicable TDs
FPT_SKP_EXT.1	Protection of TSF Data (for reading of all symmetric keys)	NDcPP	
FPT_APW_EXT.1	Protection of Administrator Passwords	NDcPP	
FPT_TST_EXT.1	TSF Testing	NDcPP	TD0836
FPT_TUD_EXT.1	Trusted Update	NDcPP	
FPT_STM_EXT.1	Reliable Time Stamps	NDcPP	
FTA_SSL_EXT.1	TSF-initiated Session Locking	NDcPP	TD0879

5.3 Functional Requirements

Table 10: Summary of SFRs

Requirement	Title
FAU_GEN.1	Audit Data Generation
FAU_GEN.2	User Identity Association
FAU_STG_EXT.1	Protected Audit Event Storage
FCS_CKM.1	Cryptographic Key Generation
FCS_CKM.2	Cryptographic Key Establishment
FCS_CKM.4	Cryptographic Key Destruction
FCS_COP.1/DataEncryption	Cryptographic Operation (AES Data Encryption/Decryption)
FCS_COP.1/SigGen	Cryptographic Operation (Signature Generation and Verification)
FCS_COP.1/Hash	Cryptographic Operation (Hash Algorithm)
FCS_COP.1/KeyedHash	Cryptographic Operation (Keyed Hash Algorithm)
FCS_NTP_EXT.1	NTP Protocol
FCS_RBG_EXT.1	Random Bit Generation
FCS_SSH_EXT.1	SSH Protocol
FCS_SSHC_EXT.1	SSH Protocol – Client
FCS_SSHS_EXT.1	SSH Protocol – Server
FCS_TLSS_EXT.1	TLS Server Protocol

Requirement	Title
FIA_AFL.1	Authentication Failure Handling
FIA_PMG_EXT.1	Password Management
FIA_UIA_EXT.1	User Identification and Authentication
FIA_UAU.7	Protected Authentication Feedback
FIA_X509_EXT.1/Rev	X.509 Certificate Validation
FIA_X509_EXT.2	X.509 Certificate Authentication
FIA_X509_EXT.3	X.509 Certificate Requests
FMT_MOF.1/ManualUpdate	Management of Security Functions Behaviour
FMT_MTD.1/CoreData	Management of TSF Data
FMT_MTD.1/CryptoKeys	Management of TSF Data
FMT_SMF.1	Specification of Management Functions
FMT_SMR.2	Restrictions on Security Roles
FPT_SKP_EXT.1	Protection of TSF Data (for reading of all symmetric keys)
FPT_APW_EXT.1	Protection of Administrator Passwords
FPT_TST_EXT.1	TSF Testing
FPT_TUD_EXT.1	Trusted Update
FPT_STM_EXT.1	Reliable Time Stamps
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
FTP_ITC.1	Inter-TSF trusted channel
FTP_TRP.1/Admin	Trusted Path

5.3.1 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 shut-down of the audit functions;

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

c. All administrative actions comprising:

- *Administrative login and logout (name of Administrator account shall be logged if individual accounts are required for Administrators).*
- *Changes to TSF data related to configuration changes (in addition to the information that a change occurred it shall be logged what has been changed).*
- *Generating/import of, changing, or deleting of cryptographic keys (in addition to the action itself a unique key name or key reference shall be logged).*
- *[Resetting passwords (name of related Administrator account shall be logged)];*

d. *Specifically defined auditable events listed in ~~**Table 2**~~* Table 11.

Table 11: Audit Events

Requirement	Auditable Events	Additional Audit Record Contents
FAU_GEN.1	None.	None.
FAU_GEN.2	None.	None.
FAU_STG_EXT.1	Configuration of local audit settings.	Identity of account making changes to the audit configuration.
FCS_CKM.1	None.	None.
FCS_CKM.2	None.	None.
FCS_CKM.4	None.	None.
FCS_COP.1/DataEncryption	None.	None.
FCS_COP.1/SigGen	None.	None.
FCS_COP.1/Hash	None.	None.
FCS_COP.1/KeyedHash	None.	None.
FCS_NTP_EXT.1	• Configuration of a new time server • Removal of configured time server	Identity if new/removed time server
FCS_RBG_EXT.1	None.	None.

Requirement	Auditable Events	Additional Audit Record Contents
FCS_SSH_EXT.1	• [Failure to establish SSH connection] • [Establishment of SSH connection] • [Termination of SSH connection session] • [Dropping of packet(s) outside defined size limits]	• [Reason for failure and Non-TOE endpoint of attempted connection (IP Address)] • [Non-TOE endpoint of connection (IP Address)] • [Non-TOE endpoint of connection (IP Address)] • [Packet size]
FCS_SSHC_EXT.1	No events specified	
FCS_SSHS_EXT.1	No events specified	
FCS_TLSS_EXT.1	Failure to establish a TLS session	Reason for failure
FIA_AFL.1	Unsuccessful login attempts limit is met or exceeded.	Origin of the attempt (e.g., IP address).
FIA_PMG_EXT.1	None.	None.
FIA_UIA_EXT.1	All use of identification and authentication mechanisms.	Provided of the attempt (e.g., IP address).
FIA_UAU.7	None.	None.
FIA_X509_EXT.1/Rev	• Unsuccessful attempt to validate a certificate • Any addition, replacement or removal of trust anchors in the TOE's trust store	• Reason for failure of certificate validation • Identification of certificates added, replaced or removed as trust anchor in the TOE's trust store
FIA_X509_EXT.2	None.	None.
FIA_X509_EXT.3	None.	None.
FMT_MOF.1/ManualUpdate	Any attempt to initiate a manual update	None.
FMT_MTD.1/CoreData	None.	None.
FMT_MTD.1/CryptoKeys	None.	None.
FMT_SMF.1	All management activities of TSF data.	None.
FMT_SMR.2	None.	None.

Requirement	Auditable Events	Additional Audit Record Contents
FPT_SKP_EXT.1	None.	None.
FPT_APW_EXT.1	None.	None.
FPT_TST_EXT.1	None.	None.
FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure)	None.
FPT_STM_EXT.1	Discontinuous changes to time - either Administrator actuated or changed via an automated process. (Note that no continuous changes to time need to be logged. See also application note on FPT_STM_EXT.1)	For discontinuous changes to time: The old and new values for the time. Origin of the attempt to change time for success and failure (e.g., IP address).
FTA_SSL_EXT.1 (if “terminate the session” is selected)	The termination of a local session by the session lock	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	• None • None • Reason for failure
FTP_TRP.1/Admin	• Initiation of the trusted path. • Termination of the trusted path. • Failure of the trusted path functions.	• None • None • Reason for failure

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 ~~(if applicable)~~, and the outcome (success or failure) of the event; and

b. For each audit event type, based on the auditable event definitions of the functional components included in the cPP/ST, *information specified in column three of* ***~~Table 2~~*** Table 11.

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_EXT.1 Protected Audit Event Storage

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

FAU_STG_EXT.1.2 The TSF shall be able to store generated audit data on the TOE itself. In addition [

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

].

FAU_STG_EXT.1.3 The TSF shall maintain a [log file] of audit records in the event that an interruption of communication with the remote audit server occurs.

FAU_STG_EXT.1.4 The TSF shall be able to store [persistent] audit records locally with a minimum storage size of [*20MB of log data with at most 9 previous log files*].

FAU_STG_EXT.1.5 The TSF shall [overwrite previous audit records according to the following rule: [*overwrite oldest record first*]] when the local storage space for audit data is full.

FAU_STG_EXT.1.6 The TSF shall provide the following mechanisms for administrative access to locally stored audit records [ability to view locally].

5.3.2 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 bits*] that meet the following: FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.3 or FIPS PUB 186-5, "Digital Signature Standard (DSS)", A.1;
- 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 or FIPS PUB 186-5, “Digital Signature Standard (DSS)”, Appendix A.2, or ISO/IEC 14888-3, “IT Security techniques -

Digital signatures with appendix - Part 3: Discrete logarithm based mechanisms", Section 6.6.;

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

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

Application Note: This SFR modified by TD0921

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 meet the following: RSAES-PKCS1-v1_5 as specified in Section 7.2 of RFC 8017, "Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.2";
- Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography";
- FFC Schemes using "safe-prime" groups that meet the following: NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" and [groups listed in RFC 7919].

] ~~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 [destruction of reference to the key directly followed by a request for garbage collection];*
- *For plaintext keys in non-volatile storage, the destruction shall be executed by the invocation of an interface provided by a part of the TSF that [*
    - *instructs a part of the TSF to destroy the abstraction that represents the key*

that meets the following: *No Standard*.

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

FCS_COP.1.1/DataEncryption The TSF shall perform *encryption/decryption* in accordance with a specified cryptographic algorithm *AES used in* [CBC, CTR,

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, CTR as specified in ISO 10116, GCM as specified in ISO 19772].

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

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

- RSA Digital Signature Algorithm,
- Elliptic Curve Digital Signature Algorithm

]

and cryptographic key sizes [

- For RSA: *[2048],*
- For ECDSA: *[256, 384, 521]*

]

that meet the following: [

- For RSA schemes: FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Section 5.5, using PKCS #1 v2.1 or FIPS PUB 186-5, "Digital Signature Standard (DSS)", Section 5.4 using PKCS #1 v2.2 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 implementing [P-256] curves that meet the following: FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Section 6 and Appendix D, Implementing “NIST Recommended" curves; or FIPS PUB 186-5, "Digital Signature Standard (DSS)", Section 6 and NIST SP 800-186 Section 3.2.1, Implementing Weierstrass curves; or ISO/IEC 14888-3, "IT Security techniques - Digital signatures with appendix - Part 3: Discrete logarithm based mechanisms", Section 6.6.]

].

Application Note: This SFR modified by TD0921

FCS_COP.1/Hash Cryptographic Operation (Hash Algorithm)

FCS_COP.1.1/Hash The TSF shall perform *cryptographic hashing services* in accordance with a specified cryptographic algorithm [SHA-1, SHA-256, SHA-384, SHA-512] ~~and cryptographic key sizes [assignment: cryptographic key sizes]~~ and **message digest sizes [160, 256, 384, 512] bits** that meet the following: *ISO/IEC 10118-3:2004*.

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

FCS_COP.1.1/KeyedHash The TSF shall perform *keyed-hash message authentication* in accordance with a specified cryptographic algorithm [HMAC-SHA-256, HMAC-SHA-512] and cryptographic key sizes *[256,*

512] **and message digest sizes [256, 512] bits** that meet the following: *ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2"*.

FCS_NTP_EXT.1 NTP Protocol

FCS_NTP_EXT.1.1 The TSF shall use only the following NTP version(s) [NTP v4 (RFC 5905)].

FCS_NTP_EXT.1.2 The TSF shall update its system time using [

- Authentication using [SHA256] as the message digest algorithm(s);

].

FCS_NTP_EXT.1.3 The TSF shall not update NTP timestamp from broadcast and/or multicast addresses.

FCS_NTP_EXT.1.4 The TSF shall support configuration of at least three (3) NTP time sources in the Operational Environment.

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 [HMAC_DRBG [SHA-512], CTR_DRBG (AES)].

FCS_RBG_EXT.1.2 The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [[*one*] software-based noise source, [*one*] platform-based noise source]] with a minimum of [256 bits] of entropy at least equal to the greatest security strength, according to ISO/IEC 18031:2011 Table C.1 "Security Strength Table for Hash Functions", of the keys and hashes that it will generate.

FCS_SSH_EXT.1 SSH Protocol

FCS_SSH_EXT.1.1 The TOE shall implement SSH acting as a [client, server] in accordance with that complies with RFCs 4251, 4252, 4253, 4254, [4344, 5656, 6668, 8268, 8308, 8332] and [*no other standard*].

FCS_SSH_EXT.1.2 The TSF shall ensure that the SSH protocol implementation supports the following authentication methods: [

- "password" (RFC 4252),
- "publickey" (RFC 4252): [
    - ssh-rsa (RFC 4253),
    - rsa-sha2-256 (RFC 8332),
    - rsa-sha2-512 (RFC 8332),
    - ecdsa-sha2-nistp256 (RFC 5656),

    ]

] and no other methods.

FCS_SSH_EXT.1.3 The TSF shall ensure that, as described in RFC 4253, packets greater than [*256KB*] in an SSH transport connection are dropped.

FCS_SSH_EXT.1.4 The TSF shall protect data in transit from unauthorised disclosure using the following mechanisms: [

- aes128-ctr (RFC 4344),
- aes256-ctr (RFC 4344),
- aes128-gcm@openssh.com (RFC 5647),
- aes256-gcm@openssh.com (RFC 5647)

] and no other mechanisms.

FCS_SSH_EXT.1.5 The TSF shall protect data in transit from modification, deletion, and insertion using: [

- hmac-sha2-256 (RFC 6668),
- hmac-sha2-512 (RFC 6668),

] and no other mechanisms.

FCS_SSH_EXT.1.6 The TSF shall establish a shared secret with its peer using: [

- diffie-hellman-group14-sha256 (RFC 8268),
- diffie-hellman-group16-sha512 (RFC 8268),
- diffie-hellman-group18-sha512 (RFC 8268),
- ecdh-sha2-nistp256 (RFC 5656),
- ecdh-sha2-nistp384 (RFC 5656),
- ecdh-sha2-nistp521 (RFC 5656),

] and no other mechanisms.

FCS_SSH_EXT.1.7 The TSF shall use SSH KDF as defined in [

- RFC 5656 (Section 4)

] to derive the following cryptographic keys from a shared secret: *session keys*.

FCS_SSH_EXT.1.8 The TSF shall ensure that [

- a rekey of the session keys,

] occurs when any of the following thresholds are met:

- one hour connection time
- no more than one gigabyte of transmitted data, or
- no more than one gigabyte of received data.

FCS_SSHC_EXT.1 SSH Protocol - Client

FCS_SSHC_EXT.1.1 The TSF shall authenticate its peer (SSH server) using: [

- using a local database by associating each host name with a public key corresponding to the following list: [

- ssh-rsa (RFC 4253),
- rsa-sha2-256 (RFC 8332),
- rsa-sha2-512 (RFC 8332),
- ecdsa-sha2-nistp256 (RFC 5656),

]

] as described in RFC 4251 section 4.1.

FCS_SSHS_EXT.1 SSH Server Protocol

FCS_SSHS_EXT.1.1 The TSF shall authenticate itself to its peer (SSH Client) using: [

- ssh-rsa (RFC 4253),
- rsa-sha2-256 (RFC 8332),
- rsa-sha2-512 (RFC 8332),
- ecdsa-sha2-nistp256 (RFC 5656),

].

FCS_TLSS_EXT.1 TLS Server Protocol

FCS_TLSS_EXT.1.1 The TSF shall implement [TLS 1.3 (RFC 8446), TLS 1.2 (RFC 5246)] and reject all other TLS and SSL versions. The TLS implementation will support the following ciphersuites:

[

- TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268
- TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268
- 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_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492
- TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246
- TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246
- 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_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288
- TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288
- TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288
- TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288
- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289
- TLS_AES_128_GCM_SHA256
- TLS_AES_256_GCM_SHA384

] and no other ciphersuites.

FCS_TLSS_EXT.1.2 The TSF shall authenticate itself using X.509 certificate(s) using [RSA with key size [2048] bits; ECDSA over NIST curves [secp256r1, secp384r1, secp521r1] and no other curves].

FCS_TLSS_EXT.1.3 The TSF shall perform key exchange using: [

- RSA key establishment with key size [2048] bits;
- EC Diffie-Hellman key agreement over NIST curves [secp256r1, secp384r1, secp521r1] and no other curves;
- Diffie-Hellman parameters [ffdhe2048]

].

FCS_TLSS_EXT.1.4 The TSF shall support [session resumption based on session IDs according to RFC 5246 (TLS 1.2), session resumption based on session tickets according to RFC 5077 (TLS 1.2), session resumption according to RFC 8446 (TLS 1.3)].

FCS_TLSS_EXT.1.5 The TSF [provides] the ability to configure the list of supported ciphersuites as defined in FCS_TLSS_EXT.1.1.

FCS_TLSS_EXT.1.6 The TSF shall prohibit the use of the following extensions:

- Early data extension

FCS_TLSS_EXT.1.7 The TSF shall [not use PSKs].

FCS_TLSS_EXT.1.8 The TSF shall [support secure renegotiation in accordance with RFC 5746 by always including the “renegotiation_info” TLS extension in TLS 1.2 ServerHello messages].

5.3.3 Identification and Authentication (FIA)

FIA_AFL.1 Authentication Failure Handling

FIA_AFL.1.1 The TSF shall detect when an Administrator configurable positive integer within [*1-20*] unsuccessful authentication attempts occur related to *Administrators attempting to authenticate remotely* using *a password*.

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

FIA_PMG_EXT.1 Password Management

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

a. Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters: ["!", "@", "#", "$", "%", "^", "&", "*", "(", ")"];

b. Minimum password length shall be *configurable to between [8] and [128] characters.*

FIA_UIA_EXT.1 User Identification and Authentication

FIA_UIA_EXT.1.1 The TSF shall allow the following actions prior to requiring the non-TOE entity to initiate the identification and authentication process:

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

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

FIA_UIA_EXT.1.3 The TSF shall provide the following remote authentication mechanisms [SSH password, SSH public key, [*Policy Manager password*]] and [no other mechanism]. The TSF shall provide the following local authentication mechanisms [password-based].

FIA_UIA_EXT.1.4 The TSF shall authenticate any administrative user’s claimed identity according to each authentication mechanism specified in FIA_UIA_EXT.1.3.

Application Note: This SFR modified by TD0900

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/Rev X.509 Certificate Validation

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

- RFC 5280 certificate validation and certification path validation **supporting a minimum path length of three certificates**.
- The certification path must terminate with a trusted CA certificate designated as a trust anchor.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of the certificate using [Certificate Revocation List (CRL) as specified in RFC 5759 Section 5].
- 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 DTLS/TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field. o Client certificates presented for DTLS/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.
    - Client certificates presented for DTLS/TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.
    - OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.

FIA_X509_EXT.1.2/Rev The TSF shall only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.

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 as specified by RFC 2986 and be able to provide the following information in the request: public key and [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.

5.3.4 Security Management (FMT)

FMT_MOF.1/Functions Management of Security Functions Behaviour

FMT_MOF.1.1/Functions The TSF shall restrict the ability to [modify the behaviour of] the functions [transmission of audit data to an external IT entity, audit functionality when Local Audit Storage Space is full] to Security Administrators.

FMT_MOF.1/ManualUpdate Management of security functions behaviour

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

FMT_MTD.1/CoreData Management of TSF Data

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

FMT_MTD.1/CryptoKeys Management of TSF data

FMT_MTD.1.1/CryptoKeys The TSF shall restrict the ability to *manage* the *cryptographic keys* to *Security Administrators*.

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 remotely;*
- *Ability to configure the access banner;*
- *Ability to configure the remote session inactivity time before session termination;*
- *Ability to update the TOE, and to verify the updates using* digital signature *capability prior to installing those updates;*
- [
    - Ability to configure local audit behaviour (e.g. changes to storage locations for audit; changes to behaviour when local audit storage space is full, changes to local audit storage size);

- Ability to modify the behaviour of the transmission of audit data to an external IT entity;
- Ability to manage the cryptographic keys;
- Ability to configure the cryptographic functionality;
- Ability to configure the list of supported (D)TLS ciphers;
- Ability to configure NTP;
- Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors;
- Ability to generate Certificate Signing Request (CSR) and process CA certificate response;
- Ability to administer the TOE locally; Ability to configure the local session inactivity time before session termination or locking;
- Ability to configure the authentication failure parameters for FIA_AFL.1;
- Ability to manage the trusted public keys database;

]

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 remotely*

are satisfied.

5.3.5 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 administrative passwords in non-plaintext form.

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

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 verify the integrity of the TOE firmware and software;
- Prior to providing any cryptographic service and [at no other time] to verify correct operation of cryptographic implementation necessary to fulfil the TSF;
- [on-demand].

to demonstrate the correct operation of the TSF.

FPT_TST_EXT.1.2 The TSF shall respond to [all failures] by [[*making any cryptographic functionality unavailable*]].

Application Note: This SFR modified by TD0836.

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

FPT_STM_EXT.1 Reliable Time Stamps

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

FPT_STM_EXT.1.2 The TSF shall [synchronise time with an NTP server].

5.3.6 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 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 ~~user~~ **Administrator**-initiated termination of the ~~user's~~ **Administrator's** own interactive session.

FTA_TAB.1 Default TOE Access Banners

FTA_TAB.1.1 Before establishing ~~a~~ **an administrative user** session the TSF shall display **a Security Administrator-specified** advisory **notice and consent** warning message regarding ~~unauthorised~~ use of the TOE.

5.3.7 Trusted path/channels (FTP)

FTP_ITC.1 Inter-TSF trusted channel

FTP_ITC.1.1 The TSF shall **be capable of using [SSH] to** provide a **trusted** communication channel between itself and ~~another trusted IT product~~ **authorized IT entities supporting the following capabilities: audit server, [no other capabilities]** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from ~~modification or~~ disclosure **and detection of modification of the channel data.**

FTP_ITC.1.2 The TSF shall permit [the TSF] to initiate communication via the trusted channel.

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for [*audit server*].

FTP_TRP.1 /Admin Trusted Path

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

FTP_TRP.1.2 /Admin The TSF shall permit remote **Administrators** ~~users~~ to initiate communication via the trusted path.

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

5.4 Assurance Requirements

21 The TOE security assurance requirements are summarized in Table 12.

Table 12: Assurance Requirements

Assurance Class	Components	Description
Security Target Evaluation	ASE_CCL.1	Conformance Claims
	ASE_ECD.1	Extended Components Definition
	ASE_INT.1	ST Introduction
	ASE_OBJ.1	Security Objectives for the operational environment
	ASE_REQ.1	Stated Security Requirements
	ASE_SPD.1	Security Problem Definition
	ASE_TSS.1	TOE Summary Specification
Development	ADV_FSP.1	Basic Functional Specification
Guidance Documents	AGD_OPE.1	Operational User Guidance
	AGD_PRE.1	Preparative User Guidance
Life Cycle Support	ALC_CMC.1	Labelling of the TOE
	ALC_CMS.1	TOE CM Coverage
Tests	ATE_IND.1	Independent Testing - conformance
Vulnerability Assessment	AVA_VAN.1	Vulnerability survey

22 In accordance with section 7.1 of the NDcPP, the following refinement is made to ASE:

a) **ASE_TSS.1.1C Refinement:** The TOE summary specification shall describe how the TOE meets each SFR. **In the case of entropy analysis, the TSS is used in conjunction with required supplementary information on Entropy.**

6 TOE Summary Specification

23 The following describes how the TOE fulfils each SFR included in section 5.3.

6.1 Security Audit

6.1.1 FAU_GEN.1

24 The TOE generates the audit records specified at FAU_GEN.1 containing fields that include the timestamp, IP address (if applicable), action, user (if applicable) and a contextual message indicating success or failure of the action.

25 The following information is logged as a result of the Security Administrator generating/importing or deleting cryptographic keys:

a) **Generate SSH Private Key**. Action and key reference.

b) **Generate TLS Server Private Key**. Action and key reference.

26 The local audit settings claimed in FAU_STG_EXT.1 are not configurable and therefore the FAU_STG_EXT.1 audit requirements are trivially satisfied.

Application Note: This SFR is modified by TD0923

6.1.2 FAU_GEN.2

27 The TOE includes the user identity in audit events resulting from actions of identified users.

6.1.3 FAU_STG_EXT.1

28 Log files are transferred via SSH (see FCS_SSH_EXT.1 and FCS_SSHC_EXT.1) to the external syslog server. Logs are transmitted in real time.

29 Logs are stored locally in rotating log files as follows:

a) **/var/log log files.** Logs are rotated weekly and up to 4 weeks of logs are kept before being removed.

b) **ssg log file.** up to 20MB of log data is kept until they are rotated. A total of 9 previous logs are kept (plus the live log).

30 Logs are overwritten by removing the oldest records first.

31 Only authorized administrators may view audit records and no capability to modify or delete the audit records is provided.

6.2 Cryptographic Support

6.2.1 FCS_CKM.1

32 The TOE supports key generation for the following asymmetric schemes:

a) **RSA Scheme.** Key sizes of 2048 used in SSH and 2048-bit used in TLS communications.

b) **ECC P-256, P-384, P-521.** P-256 is used in SSH authentication and key exchange and P-256, P-384, P-521 used in TLS.

c) **FFC Safe Primes.** Used in SSH key exchange and TLS.

33 The OpenSSL cryptographic module is implemented when generating SSH keys and the BCFIPS 2.0 (Bouncy Castle FIPS Java API) cryptographic module is implemented when generating TLS keys.

6.2.2 FCS_CKM.2

34 The TOE supports the following key establishment schemes:

a) **RSA schemes.** Used in SSH and TLS communications.

b) **ECC schemes.** Used in SSH key exchange and TLS. TOE is both sender and receiver.

c) **FFC schemes using safe primes.** Used in SSH key exchange and TLS. TOE is both sender and receiver. The following Diffie Helman groups are supported for SSH:

i) Group 14 per RFC 3526 section 3

ii) Group 16 per RFC 3526 section 5

iii) Group 18 per RFC 3526 section 7

35 The OpenSSL cryptographic module is implemented in SSH communications and the BCFIPS 2.0 (Bouncy Castle FIPS Java API) cryptographic module is implemented in TLS communications.

36 Table 13 below identifies the scheme being used by each service.

Table 13: Key Agreement Mapping

Scheme	SFR	Service
RSA Schemes	FCS_TLSS_EXT.1	Administration
	FCS_SSHS_EXT.1	Administration
	FCS_SSHC_EXT.1	Audit Server
ECC	FCS_SSHS_EXT.1	Administration
	FCS_SSHC_EXT.1	Audit Server
	FCS_TLSS_EXT.1	Administration
FFC Safe Primes	FCS_SSHS_EXT.1	Administration
	FCS_SSHC_EXT.1	Audit Server
	FCS_TLSS_EXT.1	Administration

6.2.3 FCS_CKM.4

37 Table 15 shows the origin, storage location and destruction details for cryptographic keys. Unless otherwise stated, the keys are generated by the TOE.

6.2.4 FCS_COP.1/DataEncryption

38 The TOE provides symmetric encryption and decryption capabilities using 128 and 256 bit AES in CTR, CBC and GCM mode. AES is implemented in SSH.

39 The relevant NIST CAVP certificate numbers are listed Table 4.

6.2.5 FCS_COP.1/SigGen

40 The TOE provides cryptographic signature generation and verification services using:

a) RSA Signature Algorithm with key size of 2048 bits in SSH and in TLS

b) Elliptic Curve Digital Signature Algorithm with key sizes of 256 in SSH, and 256, 384 and 521 for TLS.

41 The RSA signature verification services are used in the SSH and TLS protocol and TOE firmware integrity checks.

42 The ECDSA signature verification services are used in the SSH and TLS protocol.

43 The relevant NIST CAVP certificate numbers are listed in Table 4.

44 The OpenSSL cryptographic module is implemented in SSH communications and TOE firmware integrity checks.

45 The BCFIPS 2.0 (Bouncy Castle FIPS Java API) cryptographic module is implemented in TLS communications.

6.2.6 FCS_COP.1/Hash

46 The TOE provides cryptographic hashing services using SHA-1, SHA-256, SHA-384 and SHA-512.

47 SHA is implemented in the following parts of the TSF:

a) SSH;

b) Digital signature verification as part of trusted update validation; and

c) Hashing of passwords in non-volatile storage.

48 The relevant NIST CAVP certificate numbers are listed in Table 4.

6.2.7 FCS_COP.1/KeyedHash

49 The TOE provides keyed-hashing message authentication services using HMAC-SHA-256, and HMAC-SHA-512.

50 HMAC is implemented in SSH.

51 The characteristics of the HMACs used in the TOE are given in Table 14.

Table 14: HMAC Characteristics

Algorithm	Block Size	Key Size	Digest Size
HMAC-SHA-256	512 bits	256 bits	256 bits
HMAC-SHA-512	1024 bits	512 bits	512 bits

52 The relevant NIST CAVP certificate numbers are listed in Table 4.

6.2.8 FCS_NTP_EXT.1

53 The TOE supports NTPv4 using SHA-256 authentication. The TOE allows configuration of up to 3 NTP servers. The TOE uses pre-shared keys for authentication and integrity of the NTP server when synchronizing the time.

6.2.9 FCS_RBG_EXT.1

54 The TOE contains two cryptographic modules, each seeded by their own entropy source.

55 The OpenSSL module implements a CTR_DRBG that is seeded from a platform provided entropy source, Intel RNG. Entropy from the noise is conditioned and used to seed the DRBG with 384 bits of full entropy.

56 The BCFIPS 2.0 (Bouncy Castle FIPS Java API) Java cryptographic module implements an HMAC_DRBG that is seeded from a software provided entropy source, Jitter Entropy. Entropy from the noise is conditioned and used to seed the DRGB with 512 bits of full entropy.

57 Additional detail is provided in the proprietary Entropy Description.

6.2.10 FCS_SSH_EXT.1

58 The TOE implements SSH in compliance with RFCs 4251, 4252, 4253, 4254, 4344, 5656, 6668, 8268, 8308 section 3.1 and 8332.

59 The TOE supports password-based or public key authentication (ssh-rsa, rsa-sha2-256, rsa-sha2-512, ecdsa-sha2-nistp256). In the case of public keys, the TOE authenticates the identity of the SSH client using a local database associating authorized hosts with its corresponding public key.

60 The TOE supports public key authentication with the following algorithms, rsa-sha2-512, ecdsa-sha2-nistp256 when authenticating to an external SSH servers.

61 The TOE supports the following host key algorithms, ssh-rsa, rsa-sha2-256, rsa-sha2-512, ecdsa-sha2-nistp256.

62 The TOE authenticates the identity of remote SSH servers using a local database associating authorized hosts with its corresponding public key. The TOE supports ssh-rsa, rsa-sha2-256, rsa-sha2-512, ecdsa-sha2-nistp256 host key algorithms to associate server identity when authenticating external SSH servers.

63 The TOE examines the size of each received SSH packet. If the packet is greater than 256 KB, it is automatically dropped.

64 The TOE utilises AES-CTR-128, AES-CTR-256, AES-GCM-128 and AES-GCM-256 for SSH encryption.

65 The TOE provides data integrity for SSH connections via HMAC-SHA2-256 and HMAC-SHA2-512.

66 The TOE supports diffie-hellman-group14-sha256, diffie-hellman-group16-sha512, diffie-hellman-group18-sha512, ecdh-sha2-nistp256, ecdh-sha2-nistp384 and ecdh-sha2-nistp521 for SSH key exchanges.

67 The TOE supports the use of SSH KDF as defined in RFC 5656 (Section 4) to derive the following cryptographic keys from a shared secret: session keys.

68 The TOE will re-key SSH connections after 30 minutes or after an aggregate of 512 megabytes of data has been exchanged (whichever occurs first).

6.2.11 FCS_TLSS_EXT.1

69 The TOE accepts TLS 1.2, TLS 1.3 and rejects all other TLS and SSL versions.

70 The TOE restricts TLS to the following ciphersuites:

a) TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268

b) TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268

c) TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268

d) TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268

e) TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492

f) TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492

g) TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492

h) TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492

i) TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246

j) TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246

k) TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256 as defined in RFC 5246

l) TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246

m) TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288

n) TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288

o) TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288

p) TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288

q) TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289

r) TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289

s) TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289

t) TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289

u) TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289

v) TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289

w) TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289

x) TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289

y) TLS_AES_126_GCM_SHA256

z) TLS_AES_256_GCM_SHA384

71 Ciphersuites are user configurable.

72 The TOE performs key establishment for TLS using RSA with key size of 2048-bit, Diffie-hellman group ffdhe2048, and ECDHE curves secp256r1, secp384r1, secp521r1 and no other curves.

73 The TOE supports TLS 1.2 session resumption through the use of session ID’s according to RFC 5246, session Tickets that adhere to the structural format provided in Section 4 of RFC 5077 and TLS 1.3 session resumption according to RFC 8446.

74 Session tickets are encrypted using AES-GCM symmetric algorithms, using key size of 128 consistent with FCS_COP.1/DataEncryption.

75 Session resumption via Session ID and Tickets exist within their own contexts.

76 Session resumption with Session Tickets occurs when the TOE is presented with a Session Ticket in the Client Hello. If the Ticket is accepted an abbreviated handshake is performed and session resumed. If the Ticket is rejected, a full handshake is performed and a new session established.

77 Session resumption with Session IDs occurs when the TOE is presented with the Session ID of the session to be resumed. The TOE checks its cache for a matching ID and resumes if found. If a session ID cannot be found, a new session is established instead.

78 The TOE does not permit the use of PSKs.

79 The TOE supports TLS 1.2 secure renegotiation through the use of the renegotiation_info extension in accordance with RFC 5746 and rejects TLS 1.3 renegotiation attempts.

6.3 Identification and Authentication

6.3.1 FIA_AFL.1

80 The TOE is capable of tracking authentication failures of remote administrators.

81 When a user account has sequentially failed authentication the configured number of times the account will be locked for a Security Administrator defined time period.

82 The local console does not implement the lockout mechanism.

6.3.2 FIA_PMG_EXT.1

83 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 “!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, “)”.

84 The minimum password length is configurable by the Administrator and can range from 8 to 128 characters.

6.3.3 FIA_UIA_EXT.1

85 The TOE requires all users to be successfully identified and authenticated. The TOE warning banner is displayed prior to authentication.

86 Administrative access to the TOE is facilitated through several interfaces:

a) **CLI.** Administrative CLI via virtual serial connection.

b) **SSH CLI.** Administrative CLI via SSH.

c) **Policy Manager.** Administrative client application via TLS.

87 No administrative access is permitted until an administrator is successfully identified and authenticated.

88 The TOE warning banner is displayed prior to authentication and TOE storage services are available.

89 The TOE prompts the user to enter a username and password when accessing the CLI, SSH or Policy Manager. SSH also supports the use of public keys.

90 The TOE compares submitted passwords to the stored representation for the provided username. If there is a match and the user account is not locked (per FIA_AFL.1) a successful logon occurs. For public keys, the TOE compares the presented key with the stored user key, if the association is valid, a successful logon occurs.

6.3.4 FIA_UAU.7

91 For all authentication at the local CLI the TOE provides no feedback when the administrative password is entered so that the password is obscured.

6.3.5 FIA_X509_EXT.1/Rev

92 The TOE performs X.509 certificate validation at the following points:

a) TOE Policy Manager validation of server X.509 certificates;

b) When certificates are loaded into the TOE, such as when importing CAs, certificate responses and other device-level certificates

c) In all scenarios, certificates are checked for several validation characteristics:

d) If the certificate 'notAfter' date is in the past, then this is an expired certificate which is considered invalid;

e) The certificate chain must terminate with a trusted CA certificate;

f) Server certificates consumed by the TOE must have a 'serverAuthentication' extendedKeyUsage purpose;

g) The TOE validates a certificate path and treats a certificate as a CA certificate when certificates include the basicConstraints extensions and that the CA flag is set to "TRUE" for all CA certificates.

93 Certificate revocation checking for the above scenarios is performed using CRLs. Revocation checking is performed by the TOE by retrieving the CRL file from the URL present in the CRL Distribution point of the server and intermediate certificates.

94 As X.509 certificates are not used for trusted updates, firmware integrity self-tests or client authentication, the code-signing and clientAuthentication purpose is not checked in the extendedKeyUsage for related certificates.

95 The TOE ensures that the X.509 certificates adhere to RFC 5280 Section 6.3 (certificate validation and certificate path validation), which can be summarized as follows:

a) The public key algorithm and parameters are checked

b) The current date/time is checked against the validity period revocation status is checked

c) Issuer name of X matches the subject name of X+1

d) Name constraints are checked

e) Policy OIDs are checked

f) Policy constraints are checked; issuers are ensured to have CA signing bits

g) Path length is checked

h) Critical extensions are processed

96 If, during the entire trust chain verification activity, any certificate under review fails a verification check, then the entire trust chain is deemed untrusted.

6.3.6 FIA_X509_EXT.2

97 The TOE has a trust store where root CA and intermediate CA certificates can be stored. The trust store is not cached: if a certificate is deleted, it is immediately untrusted. If a certificate is added to the trust store, it is immediately trusted for its given scope. The use of the trust store is restricted to Security Administrators.

98 Instructions for configuring the trusted IT entities to supply appropriate X.509 certificates are captured in the guidance documents.

99 As part of the verification process, a CRL route is used to determine whether the certificate is revoked or not. If the validity of the certificate cannot be established, the validation will pass.

6.3.7 FIA_X509_EXT.3

100 The TOE generates Certificate Requests that provide public key, Common Name, Organization, Organizational Unit and Country information.

101 The TOE validates the chain of certificates from the Root CA when receiving the CA Certificate Response.

6.4 Security Management

6.4.1 FMT_MOF.1/Functions

102 The TOE restricts the ability to modify the behaviour of the transmission of audit data to an external IT entity and audit functionality when Local Audit Storage Space is full to Security Administrators.

103 Modifying the behaviour of the transmission of audit data to an external IT entity is performed when the Security Administrator configures the parameters for the external audit server.

104 Modification of audit functionality when Local Audit Storage Space is full is performed when the Security Administrator configures the parameters for log rotation as outlined in FAU_STG_EXT.1.

6.4.2 FMT_MOF.1/ManualUpdate

105 The TOE restricts the ability to perform software updates to Security Administrators.

6.4.3 FMT_MTD.1/CoreData

106 Users are required to login before being provided with access to any administrative functions.

6.4.4 FMT_MTD.1/CryptoKeys

107 The TOE restricts management of cryptographic keys to Security Administrators.

6.4.5 FMT_SMF.1

108 The TOE provides the following management capabilities:

a) Ability to administer the TOE locally (virtual serial) and remotely (SSH and Policy Manager)

b) Ability to configure the access banner via CLI, SSH CLI.

c) Ability to configure the session inactivity time before session termination

   i) The CLI / SSH CLI timeout value is set via the CLI, SSH CLI.

d) Ability to update the TOE and to verify the updates via CLI or SSH CLI.

e) Ability to configure local audit behaviour (e.g. changes to storage locations for audit; changes to behaviour when local audit storage space is full; changes to local audit storage size)

f) Ability to modify the behaviour of the transmission of audit data to an external IT entity.

g) Ability to manage the cryptographic keys (generating, importing, modifying, and deleting SSH keys) via CLI or SSH CLI

h) Ability to manage the cryptographic keys (generating, importing, modifying, and deleting X509 keys) via Policy Manager

i) Ability to configure the cryptographic functionality (SSH configuration and X509 configuration) via CLI or SSH CLI.

j) Ability to configure the cryptographic functionality (X509 configuration) via Policy Manager.

k) Ability to configure the list of supported TLS ciphers via Policy Manager.

l) Ability to configure NTP via the CLI or SSH CLI

m) Ability to manage the TOE's trust store and designate X509 certificates as trust anchors via Policy Manager.

n) Ability to generate Certificate Singing Request (CSR) and process CA certificate response via Policy Manager.

o) Ability to administer the TOE locally (virtual serial). Ability to configure the local session inactivity time before session termination via CLI or SSH.

p) Ability to configure the authentication failure parameters for FIA_AFL.1 via CLI, SSH CLI or Policy Manager.

q) Ability to manage the trusted public keys database via CLI or SSH.

6.4.6 FMT_SMR.2

109 The following user accounts are available, which are all Security Administrators:

a) **ssgconfig.** This account is used to access the CLI and SSH CLI.

b) **ssgadmin.** This account is used to access the CLI and SSH CLI

c) **Admin.** This account is used to access the TOE via Policy Manager.

110 Management of TSF data is restricted to Security Administrators.

111 All user accounts are considered Security Administrators and perform different management activities based on the available actions at each interface. The ssgconfig user is presented with a custom shell when authenticated, the ssgadmin user is presented with a Linux shell with limited access.

6.5 Protection of the TSF

6.5.1 FPT_SKP_EXT.1

112 Keys are protected as described in Table 15. In all cases, plaintext keys cannot be viewed through an interface designed specifically for that purpose.

Table 15: Keys

Key	Algorithm	Storage	Zeroization
SSH Private Keys	ECDSA, RSA	Flash - plaintext	Stored in a protected file on the Gateway OS with root access only. The administrator may zeroize this key using the shred command. This causes a three pass overwrite of the file holding the key.

Key	Algorithm	Storage	Zeroization
SSH Ephemeral Keys	AES / DH / ECDH	RAM – plaintext	OpenSSL ensures that keys (including re-keyed keys) are overwritten with zeroes when no longer required.
NTP Key	SHA-256	Flash - plaintext	Keys are destroyed when generating new keys by deleting the previous file and creating a new file. Initiated via CLI command by the Security Administrator.
TLS Server Private Keys	ECDSA, RSA	Encrypted PKCS#12	Stored in encrypted PKCS#12 keystore in the Internal DB using AES-256 (CBC). Keys are overwritten with zeroes when deleted.
TLS Server Ephemeral Keys	ECDHE, DHE, RSA	RAM - plaintext	BCFIPS 2.0 (Bouncy Castle FIPS Java API) ensures that keys are overwritten with zeroes when no longer required.

6.5.2 FPT_APW_EXT.1

113 Passwords are protected as describe in Table 16. In all cases plaintext passwords cannot be viewed through an interface designed specifically for that purpose.

Table 16: Passwords

Key/Password	Generation/ Algorithm	Storage
Locally stored administrator passwords	User generated	Flash - SHA-512 hash Stored with the standard Linux password mechanism.
TLS administrator passwords	User generated	Stored in encrypted fields in the Internal DB using AES-256 (CBC). TOE logic prevents display of plaintext passwords and PEM keys to users.

6.5.3 FPT_TST_EXT.1

114 At startup, the TOE undergoes the following tests:

a) Image verification and integrity validation.

b) BCFIPS 2.0 (Bouncy Castle FIPS Java API) cryptographic self-tests.

c) OpenSSL cryptographic module self-tests

The Administrator can also perform integrity validation tests manually via SSH or local CLI.

115 These tests ensure the correct operation of the cryptographic functionality of the TOE and verify that the correct TOE image is being used. The cryptographic functionality will not be available if the tests fail, and any operation of the TOE

supported by this functionality will not be available. When the device completes the boot up operation, this is evidence that the self-tests have passed, and that the TOE, and the cryptographic functions are operating correctly.

6.5.4 FPT_TUD_EXT.1

116 The current firmware version may be queried using the CLI or SSH CLI.

117 The Security Administrator manually initiates TOE updates from the CLI or SSH CLI. TOE update files must first be copied to the TOE and then using the “Patch Management” menu, select “Upload” then “Install” via the appropriate menu options.

118 TOE update files are digitally signed (RSA) and the signature is verified using a hardcoded public key prior to installation of the update. If verification fails, the update is aborted, and an error message is displayed. If the update succeeds, a message indicating the TOE must be rebooted to apply all changes will appear.

6.5.5 FPT_STM_EXT.1

119 The TOE makes use of NTP to maintain date and time.

120 The TOE makes use of time for the following:

a) Audit record timestamps

b) Session timeouts (lockout enforcement)

c) Certificate Expiration Validation.

d) Cryptographic functions.

6.6 TOE Access

6.6.1 FTA_SSL_EXT.1

121 The Security Administrator may configure the TOE to terminate an inactive local interactive session following a specified period of time. This is applicable to the local CLI.

6.6.2 FTA_SSL.3

122 The Security Administrator may configure the TOE to terminate an inactive remote interactive session following a specified period of time. This is applicable to the CLI, SSH CLI and Policy Manager.

6.6.3 FTA_SSL.4

123 Administrative users may terminate their own sessions at any time.

124 Remote administrators may terminate their Policy Manager sessions by clicking either “**Disconnect**” or “**Exit**”. Administrators authenticated via SSH may terminate their sessions via executing “`logout`” (command for ssgadmin) or selecting “`X) Exit (no reboot)`” (command for ssgconfig).

125 Local administrators may terminate their sessions from the Virtual Console via executing “`logout`” (command for ssgadmin) or selecting “`X) Exit (no reboot)`” (command for ssgconfig).

6.6.4 FTA_TAB.1

126 The TOE displays an administrator configurable message to users prior to login at the CLI , SSH CLI and Policy Manager.

6.7 Trusted Path/Channels

6.7.1 FTP_ITC.1

127 The TOE supports secure communication with the following IT entities:

a) Audit server per FCS_SSH_EXT.1 and FCS_SSHC_EXT.1

6.7.2 FTP_TRP.1/Admin

128 The TOE provides the following trusted paths for remote administration:

a) **SSH.** Administrative CLI via SSH per FCS_SSH_EXT.1 and FCS_SSHS_EXT.1.

b) **Policy Manager.** Administrative client application via TLS per FCS_TLSS_EXT.1.

7 Rationale

7.1 Conformance Claim Rationale

129 The following rationale is presented with regard to the PP conformance claims:

a) **TOE type.** As identified in section 2.1, the TOE is network device, consistent with the NDcPP.

b) **Security problem definition.** As shown in section 3, the threats, OSPs and assumptions are reproduced directly from the NDcPP.

c) **Security objectives.** As shown in section 4, the security objectives are reproduced directly from the NDcPP.

d) **Security requirements.** As shown in section 5, the security requirements are reproduced directly from the NDcPP. No additional requirements have been specified.

7.2 Security Objectives Rationale

130 All security objectives are drawn directly from the NDcPP.

7.3 Security Requirements Rationale

131 All security requirements are drawn directly from the NDcPP. Table 17 presents a mapping between threats and SFRs as presented in the NDcPP.

Table 17: NDcPP SFR Rationale

Identifier	SFR Rationale
T.UNAUTHORIZED_ADMINISTRATOR_ACCESS	• The Administrator role is defined in FMT_SMR.2 and the relevant administration capabilities are defined in FMT_SMF.1 and FMT_MTD.1/CoreData, with optional additional capabilities in FMT_MOF.1/Services and FMT_MOF.1/Functions • The actions allowed before authentication of an Administrator are constrained by FIA_UIA_EXT.1, and include the advisory notice and consent warning message displayed according to FTA_TAB.1 • The requirement for the Administrator authentication process is described in FIA_UIA_EXT.1 • Locking of Administrator sessions is ensured by FTA_SSL_EXT.1 (for local sessions), FTA_SSL.3 (for remote sessions), and FTA_SSL.4 (for all interactive sessions) • The secure channel used for remote Administrator connections is specified in FTP_TRP.1/Admin • (Malicious actions carried out from an Administrator session are separately addressed by T.UNDETECTED_ACTIVITY) • If the TOE provides remote administration using a password-based authentication mechanism, FIA_AFL.1

Identifier	SFR Rationale
	provides actions on reaching a threshold number of consecutive password failures.
T.WEAK_CRYPTOGRAPHY	• Requirements for key generation and key distribution are set in FCS_CKM.1 and FCS_CKM.2 respectively • Requirements for use of cryptographic schemes are set in FCS_COP.1/DataEncryption, FCS_COP.1/SigGen, FCS_COP.1/Hash, and FCS_COP.1/KeyedHash • Requirements for random bit generation to support key generation and secure protocols (see SFRs resulting from T.UNTRUSTED_COMMUNICATION_CHANNELS) are set in FCS_RBG_EXT.1 • Management of cryptographic functions is specified in FMT_SMF.1
T.UNTRUSTED_COMMUNICATION_CHANNELS	• The general use of secure protocols for identified communication channels is described at the top level in FTP_ITC.1 and FTP_TRP.1/Admin; for distributed TOEs the requirements for inter-component communications are addressed by the requirements in FPT_ITT.1 • Requirements for the use of secure communication protocols are set for allowed protocols in FCS_DTLSC_EXT.1, FCS_DTLSC_EXT.2, FCS_DTLSS_EXT.1, FCS_DTLSS_EXT.2, FCS_HTTPS_EXT.1, FCS_IPSEC_EXT.1, FCS_TLSC_EXT.1, FCS_TLSC_EXT.2, FCS_TLSS_EXT.1, FCS_TLSS_EXT.2 • Requirements for the use of secure communication protocols implemented by the packages specified in Section 2.2 may be found in the respective package's document. • Optional and selection-based requirements for use of public key certificates to support secure protocols are defined in FIA_X509_EXT.1, FIA_X509_EXT.2, FIA_X509_EXT.3
T.WEAK_AUTHENTICATION_ENDPOINTS	• The use of appropriate secure protocols to provide authentication of endpoints (as in the SFRs addressing T.UNTRUSTED_COMMUNICATION_CHANNELS) are ensured by the requirements in FTP_ITC.1 and FTP_TRP.1/Admin; for distributed TOEs the authentication requirements for endpoints in inter-component communications are addressed by the requirements in FPT_ITT.1 • Additional possible special cases of secure authentication during registration of distributed TOE components are addressed by FCO_CPC_EXT.1 and FTP_TRP.1/Join.

Identifier	SFR Rationale
T.UPDATE_COMPROMISE	• Requirements for protection of updates are set in FPT_TUD_EXT.1 • Additional optional use of certificate-based protection of signatures can be specified using FPT_TUD_EXT.2, supported by the X.509 certificate processing requirements in FIA_X509_EXT.1, FIA_X509_EXT.2 and FIA_X509_EXT.3 • Requirements for management of updates are defined in FMT_SMF.1 and (for manual updates) in FMT_MOF.1/ManualUpdate, with optional requirements for automatic updates in FMT_MOF.1/AutoUpdate
T.UNDETECTED_ACTIVITY	• Requirements for basic auditing capabilities are specified in FAU_GEN.1 and FAU_GEN.2, with timestamps provided according to FPT_STM_EXT.1 and if applicable, protection of NTP channels in FCS_NTP_EXT.1. • Requirements for protecting audit records stored on the TOE are specified in FAU_STG.1. • Requirements for secure storage and transmission of local audit records to an external IT entity via a secure channel are specified in FAU_STG_EXT.1 and FAU_STG_EXT.1. • Optional additional requirements for dealing with potential loss of locally stored audit records are specified in FAU_STG_EXT.2, and FAU_STG_EXT.3.
T.SECURITY_FUNCTIONALITY_COMPROMISE	• Protection of secret/private keys against compromise is specified in FPT_SKP_EXT.1 • Secure destruction of keys is specified in FCS_CKM.4 • If (optionally) management of keys is provided by the TOE then this is specified in FMT_SMF.1 and confining this functionality to Security Administrators is required by FMT_MTD.1/CryptoKeys • If optional local administration using a password-based authentication mechanism is provided by the TOE, FIA_UAU.7 provides protection of password entry by providing only obscured feedback at the local console. • If the TOE provides password-based authentication mechanisms, requirements for password lengths and available characters are set in FIA_PMG_EXT.1. Requirements for secure storage of passwords are set in FPT_APW_EXT.1
T.SECURITY_FUNCTIONALITY_FAILURE	• Requirements for running self-test(s) are defined in FPT_TST_EXT.1
P.ACCESS_BANNER	• An advisory notice and consent warning message is required to be displayed by FTA_TAB.1