Crestron DigitalMedia NVX®AV-over-IP v7.1

Security Target

Version 1.0

03 October 2024

Prepared for:

Crestron Electronics, Inc.

15 Volvo Drive
Rockleigh, New Jersey 07647

Prepared by:

leidos

Common Criteria Testing Laboratory
6841 Benjamin Franklin Drive
Columbia, Maryland 21046

Table of Contents

List of Figures and Tables

1 Security Target Introduction

This section introduces the Target of Evaluation (TOE) and provides the Security Target (ST) and TOE references, TOE overview, and TOE description. It also contains the ST and TOE conformance claims, ST conventions, glossary, and list of abbreviations.

This ST includes the following additional sections:

- Security Problem Definition (Section 2)—describes the threats and assumptions that define the security problem to be addressed by the TOE and its environment
- Security Objectives (Section 3)—describes the security objectives for the TOE and its operational environment necessary to counter the threats and satisfy the assumptions that define the security problem
- IT Security Requirements (Section 4)—specifies the security functional requirements (SFRs) and security assurance requirements (SARs) to be met by the TOE
- TOE Summary Specification (Section 5)—describes the security functions of the TOE and how they satisfy the SFRs
- Protection Profile Claims (Section 6)—provides rationale supporting the claims for conformance of the ST and the TOE to [cPPND]
- Rationale (Section 7)—provides mappings and rationale for the security problem definition, security objectives, security requirements, and security functions to justify their completeness, consistency, and suitability.

1.1 Security Target, Target of Evaluation, and Common Criteria Identification

ST Title: Crestron DigitalMedia NVX®AV-over-IP v7.1 Security Target

ST Version: Version 1.0

ST Date: 03 October 2024

TOE Identification: Crestron DigitalMedia NVX®AV-over-IP v7.1

The TOE includes the following appliance models, each with firmware version 7.1.5259.00081:

- DM-NVX-E10
- DM-NVX-E20
- DM-NVX-E20-2G
- DM-NVX-E30
- DM-NVX-E30C
- DM-NVX-E760
- DM-NVX-E760C
- DM-NVX-D10
- DM-NVX-D20
- DM-NVX-D30
- DM-NVX-D30C
- DM-NVX-D200
- DM-NVX-D80-IOAV

- DM-NVX-350
- DM-NVX-350C
- DM-NVX-351
- DM-NVX-351C
- DM-NVX-352
- DM-NVX-352C
- DM-NVX-360
- DM-NVX-360C
- DM-NVX-363
- DM-NVX-363C.

Each appliance contains an Intel Arria 10 SX SoC FPGA that includes an ARM Cortex-A9 MPCore processor implementing the ARMv7-A microarchitecture. "C" indicates that the model is a form factor with a chassis card.

TOE Developer: Crestron Electronics, Inc.

Evaluation Sponsor: Crestron Electronics, Inc.

CC Identification: Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5, April 2017.

1.2 Conformance Claims

This ST and the TOE it describes are conformant to the following CC specifications:

- Common Criteria for Information Technology Security Evaluation Part 2: Security Functional Components, Version 3.1 Revision 5, April 2017
  - Part 2 Extended
- Common Criteria for Information Technology Security Evaluation Part 3: Security Assurance Components, Version 3.1 Revision 5, April 2017
  - Part 3 Conformant.

This ST and the TOE it describes are conformant to the following Protection Profile:

- *collaborative Protection Profile for Network Devices*, Version 2.2e, 23 March 2020 ([cPPND]), including the following optional and selection-based SFRs: FAU_STG.1, FCS_HTTPS_EXT.1; FCS_NTP_EXT.1; FCS_SSHS_EXT.1; FCS_TLSC_EXT.1; FCS_TLSS_EXT.1; FIA_X509_EXT.1/Rev; FIA_X509_EXT.2; FIA_X509_EXT.3; FMT_MOF.1/Services; and FMT_MTD.1/CryptoKeys.

The following NIAP Technical Decisions are applicable to the claimed Protection Profile:

- TD0527 – Updates to Certificate Revocation Testing (FIA_X509_EXT.1)
  - This TD is applicable to the TOE but relates solely to evaluation activities so it does not affect the ST.
- TD0528 – NIT Technical Decision for Missing EAs for FCS_NTP_EXT.1.4
  - This TD is applicable to the TOE but relates solely to evaluation activities so it does not affect the ST.

- TD0536 – NIT Technical Decision for Update Verification Inconsistency
    - This TD is applicable to the TOE but relates solely to evaluation activities so it does not affect the ST.
- TD0537 – NIT Technical Decision for Incorrect reference to FCS_TLSC_EXT.2.3
    - This TD is applicable to the TOE and the ST. The TD modifies [cPPND] Application Note 113.
- TD0546 – NIT Technical Decision for DTLS - clarification of Application Note 63
    - This TD is not applicable to the TOE because DTLS is not used.
- TD0547 – NIT Technical Decision for Clarification on developer disclosure of AVA_VAN
    - This TD is applicable to the TOE.
- TD0555 – NIT Technical Decision for RFC Reference incorrect in TLSS Test
    - This TD is applicable to the TOE but relates solely to evaluation activities so it does not affect the ST.
- TD0556 – NIT Technical Decision for RFC 5077 question
    - This TD is applicable to the TOE but relates solely to evaluation activities so it does not affect the ST.
- TD0563 – NiT Technical Decision for Clarification of audit date information
    - This TD is applicable to the TOE.
- TD0564 – NiT Technical Decision for Vulnerability Analysis Search Criteria
    - This TD is applicable to the TOE but relates solely to evaluation activities so it does not affect the ST.
- TD0569 – NIT Technical Decision for Session ID Usage Conflict in FCS_DTLSS_EXT.1.7
    - This TD is applicable to the TOE but relates to application notes and evaluation activities so it does not affect the ST.
- TD0570 – NiT Technical Decision for Clarification about FIA_AFL.1
    - This TD is a clarification to the requirement and therefore is generally applicable to the TOE but does not change any of the requirements.
- TD0571 – NiT Technical Decision for Guidance on how to handle FIA_AFL.1
    - This TD is a clarification to the requirement and therefore is generally applicable to the TOE but does not change any of the requirements.
- TD0572 – NiT Technical Decision for Restricting FTP_ITC.1 to only IP address identifiers
    - This TD is a clarification to the FCS_TLSC_EXT and FTP_ITC.1 requirements and therefore is generally applicable to the TOE but does not change any of the requirements.
- TD0580 – NiT Technical Decision for clarification about use of DH14 in NDcPPv2.2e.
    - This TD is a clarification to the DH14 requirement and modifies the selections in the SFR. Therefore it is applicable to the ST.
- TD0581– NiT Technical Decision for Elliptic curve-based key establishment and NIST SP 800-56Arev3
    - This TD adds a selection for Elliptic curve-based key establishment based on NIST SP 800-56Arev3 and is therefore applicable to the ST as it has been selected.

- TD0591– NIT Technical Decision for Virtual TOEs and hypervisors
    - This TD modifies an assumption and therefore is applicable to the TOE.
- TD0592– NIT Technical Decision for Local Storage of Audit Records
    - This TD is applicable to the TOE, though it has no material effect on the ST.
- TD0631– NIT Technical Decision for Clarification of public key authentication for SSH Server
    - This TD is applicable to the TOE.
- TD0632– NIT Technical Decision for Consistency with Time Data for vNDs
    - This TD addresses consistency with time data for virtual network devices and adds a selection to FPT_STM_EXT.1. Therefore, it is applicable to the ST, although the ST does not include this selection.
- TD0635– NIT Technical Decision for TLS Server and Key Agreement Parameters
    - This TD is applicable to the TOE but relates solely to evaluation activities so it does not affect the ST.
- TD0636– NIT Technical Decision for Clarification of Public Key User Authentication for SSH
    - This TD is not applicable to the TOE because the ST does not claim FCS_SSHC_EXT.1.
- TD0638– NIT Technical Decision for Key Pair Generation for Authentication
    - This TD is a clarification to FCS_CKM.1 and therefore is generally applicable to the TOE but does not change any of the requirements.
- TD0639– NIT Technical Decision for Clarification for NTP MAC Keys
    - This TD is applicable to the TOE, though it has no material effect on the ST.
- TD0670– NIT Technical Decision for Mutual and Non-Mutual Auth TLSC Testing
    - This TD is applicable to the TOE but relates solely to evaluation activities so it does not affect the ST.
- TD0738– NIT Technical Decision for Link to Allowed-With List
    - This TD is applicable to the TOE, though it has no material effect on the ST.
- TD0790– NIT Technical Decision: Clarification Required for testing IPv6
    - This TD is applicable to the TOE but relates solely to evaluation activities so it does not affect the ST.
- TD0792– NIT Technical Decision: FIA_PMG_EXT.1 - TSS EA not in line with SFR
    - This TD is applicable to the TOE but relates solely to evaluation activities so it does not affect the ST.
- TD0800– Updated NIT Technical Decision for IPsec IKE/SA Lifetimes Tolerance
    - This TD is not applicable to the TOE because the ST does not claim FCS_IPSEC_EXT.1.

1.3 Conventions

The following conventions are used in this document:

- Security Functional Requirements—Part 1 of the CC defines the approved set of operations that may be applied to functional requirements: iteration; selection; assignment; and refinement.

  - Iteration—allows a component to be used more than once with varying operations. In this ST, the only iterated requirements are those reproduced from [cPPND], which uses descriptive strings to distinguish iterations of a requirement. For example, iterations of FCS_COP.1 are identified FCS_COP.1/DataEncryption, FCS_COP.1/SigGen, FCS_COP.1/Hash, and FCS_COP.1/KeyedHash.
  - Selection—allows the specification of one or more elements from a list. Selections completed in the ST are indicated using underlined text and are enclosed by brackets (e.g., [selection]). Where a selection is completed inside another selection, it is indicated by underlined italic text enclosed in brackets (e.g., [*embedded selection*]).
  - Assignment—allows the specification of an identified parameter. Assignments completed in the ST are indicated using bold text and are enclosed by brackets (e.g., [**assignment**]). An assignment within a selection is identified using bold underlined text enclosed by brackets (e.g., [**selected-assignment**]).
  - Refinement—allows the addition of details. Refinements made in the ST of requirements drawn from [cPPND] would be indicated using bold for additions or modifications and strike-through for deletions (e.g., "… ~~some~~ **all** objects).
- Other sections of the ST—other sections of the ST use bolding and/or different fonts (such as `Courier`) to highlight text of special interest, such as captions, commands, or filenames specific to the TOE.

1.4 Abbreviations and Acronyms

Table 1: Abbreviations and Acronyms

Abbreviation	Definition
AES	Advanced Encryption Standard
CBC	Cipher Block Chaining
DH	Diffie-Hellman
DRBG	Deterministic Random Bit Generator
DSA	Digital Signature Algorithm
ECDSA	Elliptic Curve Digital Signature Algorithm
HMAC	Hash-based Message Authentication Code
LDAP	Lightweight Directory Access Protocol
MAC	Message Authentication Code
NTP	Network Time Protocol
NVX	Shorthand for the TOE: Crestron Digital Media NVX Series
SHA	Secure Hash Algorithm
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Functionality
USB	Universal Serial Bus

1.5 TOE Overview

The TOE is Crestron DigitalMedia NVX®AV-over-IP v7.1. NVX is a series of audio & video (AV) over IP network devices that encrypt, decrypt and transmit HDMI video, USB and analog audio data over customer networks. These communication streams use an AES-based HDCP standard that is not covered by the *collaborative Protection Profile for Network Devices*, Version 2.2e, 23 March 2020 ([cPPND]), and therefore is not evaluated. The AES-based HDCP standard does not provide any TSF. The focus of this evaluation is on the TOE functionality supporting the claims in [cPPND].

The security functionality specified in [cPPND] includes protection of communications between the TOE and external IT entities, identification and authentication of administrators, auditing of security-relevant events, ability to verify the source and integrity of updates to the TOE, and use of NIST-validated cryptographic mechanisms. The [cPPND] does not define requirements for encryption of audio and video so the TOE focuses on the security of the network channels used for syslog, management, and authentication.

1.6 TOE Description

The TOE is a digital video and audio distribution network device that switches 4K video sources and displays at 60 frames per second (fps) with full 4:4:4 color sampling, High Dynamic Range (HDR) video support, standard 1-Gigabit Ethernet infrastructure, and Pixel Perfect Processing technology to provide video transport in all applications. The digital video and audio transport and encoding/decoding are not evaluated.

For the purpose of this evaluation, the TOE is treated as a network device offering NIST validated cryptographic functions, security auditing, secure administration, trusted updates, self-tests, and secure connections to other servers (e.g., to export audit records), protected using HTTPS/TLS and SSH.

Cryptographic functionality is performed by the TOE's '*Crestron Crypto Kernel for Open SSL*' software module that includes third-party SafeLogic OpenSSL in support of higher level protocols (TLS, SSH). The module's FIPS-Approved cryptographic algorithms have obtained CAVP certificates.

The TOE audits security relevant events, stores audit records locally, and can be configured to forward its audit records to an external syslog server in the network environment. An administrator can configure the TOE to solicit time from an NTP server, and alternatively the administrator can manually set the TOE's time.

The TOE uses TLS to protect communications with an external syslog server, offers a management GUI protected by TLS/HTTPS, and provides a management Command Line Interface (CLI) protected by SSH.

Administrators are able to query the current version of the product firmware and manage the security functions of the TOE, including performing updates on the product. Public/private keys are used to provide digital signatures for protection of the update files.

The TOE provides self-tests to ensure the integrity and correct operation of the TOE.

1.6.1 Physical Scope

The NVX TOE is deployed as a single physical appliance that serves as one endpoint for an audio/visual (AV) over IP (Transmission Control Protocol/Internet Protocol (TCP/IP)) connection. The NVX models are available as surface-mountable endpoints, OPS (Open Pluggable Specification) endpoints and chassis-based cards. Models are available as encoder-only, decoder-only, or combined encoder/decoder products. The NVX Models included in the TOE are identified in Section 1.1.

The NVX Models differ in terms of form factor, function (sender, receiver), number and types of external control and data ports and maximum Input/Output resolution. The NVX models use the same firmware image files and provide equivalent security-relevant functionality. There are no security relevant differences between the appliance models.

Each TOE appliance includes an Intel Arria 10 SoC FPGA that includes an ARM Cortex-A9 MPCore processor implementing the ARMv7-A microarchitecture and an Angstrom Linux v2014.12 operating system, which use version 4.19 of the Linux kernel. The operating system has been hardened; does not permit operators (even an authorized administrator) access to the OS; and includes NVX-developed firmware. Additionally, the TOE includes the following third party components:

Table 2: Third Party Components

Component	Version
lighttpd	1.4.52
Redis	5.0.14
OpenSSH	9.8p1
Net-SNMP	5.9.4
NTPsec	1.2.3
SafeLogic OpenSSL	1.0.2zd

The administrator gains local access to the CLI from a workstation directly connected to a network port and using SSH client software, and remote access to the web management interface using a browser.

The TOE in its evaluated configuration requires the following components in its operational environment:

- A TLS-protected syslog server that receives audit events from the TOE
- NTP servers with which the TOE can synchronize its clock
- OCSP Server for certificate revocation status validation
- Certificate Authority, by which the administrator can have a Certificate Signing Request (CSR) signed.
- A client workstation for administrator access to the web GUI and CLI with:
    - A supported browser:
        - Firefox®, version 31 and later
        - Internet Explorer, version 11
        - Microsoft Edge
        - Safari®, version 6 and later
        - Chrome™, version 31 and later.
    - An SSH Client for local and remote access to the CLI.

1.6.2 Logical Scope

This section summarizes the security functions provided by the TOE:

- Security audit
- Cryptographic support
- Identification and authentication
- Security management
- Protection of the TSF

- TOE access
- Trusted path/channels.

1.6.2.1 Security Audit

The TOE generates audit events associated with identification and authentication, management, updates, and user sessions. The TOE can store the events in a local log and export them to a syslog server using a TLS protected channel.

1.6.2.2 Cryptographic Support

The TOE provides CAVP certified cryptography in support of its SSH, TLS, and NTP implementations and for verifying TOE update package signatures. Cryptographic services include key management, random bit generation, symmetric encryption and decryption, digital signature, and secure hashing.

1.6.2.3 Identification and Authentication

The TOE requires users to be identified and authenticated before they can use functions mediated by the TOE, with the exception of reading the login banner. The TOE authenticates a user's credentials (password, key) using a local mechanism provided by the TOE. The TOE also provides X.509 certificate checking for its TLS connections.

1.6.2.4 Security Management

The TOE provides CLI and web-based management interfaces that an administrator can access remotely via a network port. The CLI can also be accessed locally by directly connecting to a network port and using SSH. Remote connections to the management interface are protected with SSH for the CLI and HTTPS for the GUI. The management interface is limited to the authorized administrator.

1.6.2.5 Protection of the TSF

The TOE implements various self-protection mechanisms. The TOE performs self-tests that cover the correct operation of the TOE. It provides functions necessary to securely update the TOE. It relies upon either manually provided time or an NTP server in its environment to ensure reliable timestamps. It protects sensitive data such as passwords and cryptographic keys stored on the TOE's internal Flash so that they are not accessible even by an authorized administrator.

1.6.2.6 TOE Access

The TOE will terminate local and remote interactive sessions after a configurable period of inactivity. The TOE additionally provides the capability for administrators to terminate their own interactive sessions. The TOE can be configured to display an advisory and consent warning message before establishing a user session.

1.6.2.7 Trusted Path/Channels

The TOE provides local administration which is subject to physical protection. To access the TOE locally, an administrator must directly connect their workstation to a network port and use SSH and successfully login. When accessed remotely, the CLI and GUI management interfaces are protected by SSH and TLS respectively, thus ensuring protection against modification and disclosure.

The TOE protects communications with the external syslog servers from modification and disclosure by using TLS.

1.7 TOE Documentation

The TOE is supplied with the following guidance documentation that describes the installation process for the TOE and provides guidance for configuration and secure use of its security features:

- Crestron DigitalMedia NVX® AV-over-IP v7.1 Common Criteria Evaluated Configuration Guide (CCECG), Version 1.0, October 03, 2024
- DM-NVX-350, DM-NVX-351, and DM-NVX-352 Quick Start (Doc.8391C)
- DM-NVX-350C, DM-NVX-351C, and DM-NVX-352C Quick Start (Doc. 8392B)
- DM-NVX-E10 and DM-NVX-D10 Quick Start (Doc. 9001A)
- DM-NVX-E20 and DM-NVX-D20 Quick Start (Doc. 9000A)
- DM-NVX-E20-2G Quick Start (Doc. 9160A)
- DM-NVX-E30 and DM-NVX-D30 Quick Start (Doc.8906B)
- DM-NVX-E30C/DM-NVX-D30C Quick Start (Doc. 8346A)
- DM-NVX-D80-IOAV Quick Start (Doc. 8526A)
- DM-NVX-D200 Quick Start (Doc. 9091A)
- DM-NVX-363 and DM-NVX-360 Quick Start (Doc. 8634B)
- DM-NVX-363C and DM-NVX-360C Quick Start (Doc. 8636A)
- DM-NVX-E760 Quick Start (Doc. 8646B)
- DM-NVX-E760C Quick Start (Doc. 8638B).

1.8 Excluded Functionality

The list below identifies features or protocols that are not evaluated or must be disabled, and the rationale why. Note that this does not mean the features cannot be used in the evaluated configuration (unless explicitly stated so). It means that the features were not evaluated and/or validated by an independent third party and the functional correctness of the implementation is vendor assertion. Evaluated functionality is scoped exclusively to the security functional requirements specified in this Security Target. In particular, only the following protocols implemented by the TOE have been tested, and only to the extent specified by the security functional requirements: TLS, HTTPS, SSH, NTP authentication. The features below are out of scope.

Table 3: Excluded Functionality

Feature	Description
Net-SNMP, Telnet and HTTP Management Protocols	Net-SNMP, Telnet and HTTP are disabled by default and must not be enabled in the evaluated configuration. Only SSH and HTTPS are used for the remote management protocols to manage the TOE.
External LDAP/AD authentication	Users must be authenticated using the local authentication method. External LDAP/AD must not be used.

Feature	Description
802.1X, SCIP traffic, AES-based HDCP	The TOE uses SCIP for device –to-device communication, 802.1X is used for network access, and AES based HDCP for Audio and video. The TOE is not distributed and therefore device-to-device communication and SCIP are not within scope of the evaluation. Additionally, the [cPPND] does not define requirements for these types of traffic/protocols and therefore they have not been evaluated.
Crestron XiO Cloud® service	Allows supported Crestron devices across an enterprise to be managed and configured from one central and secure location in the cloud. It is disabled in the evaluated configuration.
Crestron Toolbox™ application	Crestron Toolbox™ application is excluded from the evaluated configuration. The evaluation only includes access to the GUI via browser.
Any features not associated with SFRs in claimed [cPPND],	[cPPND] forbids adding additional requirements to the Security Target (ST). If additional functionalities or products are mentioned in the ST, it is for completeness only.

2 Security Problem Definition

This ST includes by reference the Security Problem Definition (comprising threat statements, assumptions, and organizational security policies) from [cPPND]. The PP offers additional information about the threats, assumptions, and organizational security policies, but that has not been reproduced here and the PP should be consulted if there is interest in that material.

In general, the [cPPND] has presented a Security Problem Definition appropriate for network infrastructure devices, and as such is applicable to the TOE.

3 Security Objectives

The [cPPND] defines the following security objectives for the operational environment of the TOE.

Table 4: Security Objectives for the Operational Environment

Objective	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.
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 TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are assumed to monitor the revocation status of all certificates in the TOE's trust store and to remove any certificate from the TOE’s trust store in case such certificate can no longer be trusted.
OE.UPDATES	The TOE firmware and software is updated by an Administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
OE.ADMIN_CREDENTIALS_SECURE	The Administrator’s credentials (private key) used to access the TOE must be protected on any other platform on which they reside.
OE.RESIDUAL_INFORMATION	The Security Administrator ensures that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment.

4 IT Security Requirements

This section defines the Security Functional Requirements (SFRs) and Security Assurance Requirements (SARs) that serve to represent the security functional claims for the TOE and to scope the evaluation effort.

The SFRs have all been drawn from [cPPND]. As such, operations on SFRs already performed in that PP are not identified here. Rather, the SFRs have been copied from [cPPND] and any formatting used in that PP has been removed. Operations performed on SFRs in the writing of this ST are identified in accordance with the conventions described in Section 1.3.

The SARs are the set of SARs specified in [cPPND].

4.1 Extended Requirements

All of the extended requirements in this ST have been drawn from [cPPND]. The [cPPND] defines the following extended SFRs and since they are not redefined in this ST, the [cPPND] should be consulted for more information in regards to these CC extensions.

- FAU_STG_EXT.1—Protected Audit Event Storage
- FCS_HTTPS_EXT.1 – HTTPS Protocol
- FCS_NTP_EXT.1—NTP Protocol
- FCS_RBG_EXT.1—Random Bit Generation
- FCS_SSHS_EXT.1—SSH Server Protocol
- FCS_TLSC_EXT.1—TLS Client Protocol
- FCS_TLSS_EXT.1—TLS Server Protocol
- FIA_PMG_EXT.1—Password Management
- FIA_UAU_EXT.2—Password-Based Authentication Mechanism
- FIA_UIA_EXT.1—User Identification and Authentication
- FIA_X509_EXT.1—X.509 Certificate Validation
- FIA_X509_EXT.2—X.509 Certificate Authentication
- FIA_X509_EXT.3—X.509 Certificate Requests
- FPT_APW_EXT.1—Protection of Administrator Passwords
- FPT_SKP_EXT.1—Protection of TSF Data
- FPT_STM_EXT.1—Reliable Time Stamps
- FPT_TST_EXT.1—TSF Testing
- FPT_TUD_EXT.1—Trusted Update
- FTA_SSL_EXT.1—TSF-Initiated Session Locking

4.2 TOE Security Functional Requirements

The following table identifies the SFRs that are satisfied by the TOE.

Table 5: TOE Security Functional Components

Requirement Class	Requirement Component
FAU: 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

Requirement Class	Requirement Component
FCS: Cryptographic support	FCS_CKM.1—Cryptographic Key Generation
	FCS_CKM.2—Cryptographic Key Establishment
	FCS_CKM.4—Cryptographic Key Destruction
	FCS_COP.1/DataEncryption—Cryptographic Operation (AES Data Encryption/Decryption)
	FCS_COP.1/SigGen—Cryptographic Operation (Signature Generation and Verification)
	FCS_COP.1/Hash—Cryptographic Operation (Hash Algorithm)
	FCS_COP.1/KeyedHash—Cryptographic Operation (Keyed Hash Algorithm)
	FCS_HTTPS_EXT.1 – HTTPS Protocol
	FCS_NTP_EXT.1—NTP Protocol
	FCS_RBG_EXT.1—Random Bit Generation
	FCS_SSHS_EXT.1—SSH Server Protocol
	FCS_TLSC_EXT.1—TLS Client Protocol without Mutual Authentication
	FCS_TLSS_EXT.1—TLS Server Protocol without Mutual Authentication
FIA: Identification and authentication	FIA_AFL.1—Authentication Failure Management
	FIA_PMG_EXT.1—Password Management
	FIA_UAU_EXT.2—Password-Based Authentication Mechanism
	FIA_UAU.7—Protected Authentication Feedback
	FIA_UIA_EXT.1—User Identification and Authentication
	FIA_X509_EXT.1/Rev—X.509 Certificate Validation
	FIA_X509_EXT.2—X.509 Certificate Authentication
	FIA_X509_EXT.3—X.509 Certificate Requests
FMT:Security Management	FMT_MOF.1/ManualUpdate—Management of Security Functions Behavior
	FMT_MOF.1/Services—Management of Security Functions Behavior
	FMT_MTD.1/CoreData—Management of TSF Data
	FMT_MTD.1/CryptoKeys—Management of TSF Data
	FMT_SMF.1—Specification of Management Functions
	FMT_SMR.2—Restrictions on Security Roles
FPT: Protection of the TSF	FPT_APW_EXT.1—Protection of Administrator Passwords
	FPT_SKP_EXT.1—Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)
	FPT_STM_EXT.1—Reliable Time Stamps
	FPT_TST_EXT.1—TSF Testing
	FPT_TUD_EXT.1—Trusted update
FTA: TOE access	FTA_SSL_EXT.1—TSF-Initiated Session Locking
	FTA_SSL.3—TSF-Initiated Termination

Requirement Class	Requirement Component
	FTA_SSL.4—User-Initiated Termination
	FTA_TAB.1—Default TOE Access Banners
FTP: Trusted path/channels	FTP_ITC.1—Inter-TSF Trusted Channel
	FTP_TRP.1/Admin—Trusted Path

4.2.1 Security Audit (FAU)

4.2.1.1 Audit Data Generation (FAU_GEN.1)

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

a) Start-up and shut-down of the audit functions;

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

c) All administrative actions comprising:

- Administrative login and logout (name of user account shall be logged if individual user accounts are required for Administrators).
- Changes to TSF data related to configuration changes (in addition to the information that a change occurred it shall be logged what has been changed).
- Generating/import of, changing, or deleting of cryptographic keys (in addition to the action itself a unique key name or key reference shall be logged).
- Resetting passwords (name of related user account shall be logged).
- [no other actions];

d) Specifically defined auditable events listed in Table **6**.

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

a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event; and

b) For each audit event type, based on the auditable event definitions of the functional components included in the cPP/ST, information specified in column three of Table **6**.

Table 6: Security Functional Requirements and Auditable Events

Requirement	Auditable Events	Additional Audit Record Contents
FAU_GEN.1	None.	None.
FAU_GEN.2	None.	None.
FAU_STG.1	None.	None.
FAU_STG_EXT.1	None.	None.
FCS_CKM.1	None.	None.
FCS_CKM.2	None.	None.
FCS_CKM.4	None.	None.
FCS_COP.1/DataEncryption	None.	None.
FCS_COP.1/SigGen	None.	None.
FCS_COP.1/Hash	None.	None.
FCS_COP.1/KeyedHash	None.	None.

Requirement	Auditable Events	Additional Audit Record Contents
FCS_HTTPS_EXT.1	Failure to establish a HTTPS session	Reason for failure
FCS_NTP_EXT.1	• Configuration of a new time server • Removal of configured time server	Identity of new/removed time server
FCS_RBG_EXT.1	None.	None.
FCS_SSHS_EXT.1	Failure to establish an SSH session	Reason for failure
FCS_TLSC_EXT.1	Failure to establish a TLS session	Reason for failure
FCS_TLSS_EXT.1	Failure to establish a TLS session	Reason for failure
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 mechanism.	Origin of the attempt (e.g., IP address).
FIA_UAU_EXT.2	All use of identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UAU.7	None.	None.
FIA_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_MOF.1/Services	None.	None.
FMT_MTD.1/CoreData	None.	None.
FMT_MTD.1/CryptoKeys	None.	None.
FMT_SMF.1	All management activities of TSF data.	None.
FMT_SMR.2	None.	None.
FPT_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).

Requirement	Auditable Events	Additional Audit Record Contents
FTA_SSL_EXT.1 (if "terminate the session" is selected)	The termination of a local session by the session locking mechanism.	None.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	None.
FTA_SSL.4	The termination of an interactive session.	None.
FTA_TAB.1	None.	None.
FTP_ITC.1	• Initiation of the trusted channel. • Termination of the trusted channel. • Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempt.
FTP_TRP.1/Admin	• Initiation of the trusted path. • Termination of the trusted path. • Failure of the trusted path functions.	None.

4.2.1.2 User Identity Association (FAU_GEN.2)

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.

4.2.1.3 Protected Audit Trail Storage (FAU_STG.1)

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

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

4.2.1.4 Protected Audit Event Storage (FAU_STG_EXT.1)

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

FAU_STG_EXT.1.2 The TSF shall be able to store generated audit data on the TOE itself. In addition [The TOE shall consist of a single standalone component that stores audit data locally].

FAU_STG_EXT.1.3 The TSF shall [overwrite previous audit records according to the following rule: **[overwriting the oldest log record first]**] when the local storage space for audit data is full.

4.2.2 Cryptographic Support (FCS)

4.2.2.1 Cryptographic Key Generation (FCS_CKM.1)

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;
- FFC Schemes using ‘safe-prime’ groups that meet the following: “NIST Special Publication 800-56A Revision 3, Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography” and [*RFC 3526*]

].

4.2.2.2 Cryptographic Key Establishment (FCS_CKM.2)

FCS_CKM.2.1[1] The TSF shall perform cryptographic key establishment in accordance with a specified cryptographic key establishment method: [

- Elliptic curve-based key establishment schemes that meets the following: NIST Special Publication 800-56A Revision 3, “Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography”;
- Finite field-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 2, “Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography”;
- FFC Schemes using “safe-prime” groups that meet the following: ‘NIST Special Publication 800-56A Revision 3, “Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography” and [*groups listed in RFC 3526*[2]]

].

4.2.2.3 Cryptographic Key Destruction (FCS_CKM.4)

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-pass*] overwrite consisting of [*a new value of the key*]]***;***

  that meets the following: No Standard.

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

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

[1] Modified by TD0581 to allow for selection of Elliptic curve-based key establishment schemes that meet NIST Special Publication 800-56A Revision 3.

[2] Modified by TD0580 to clarify requirements for RFCs are restricted to groups.

as specified in ISO 18033-3, [CBC as specified in ISO 10116, CTR as specified in ISO 10116, GCM as specified in ISO 19772].

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

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 and cryptographic key sizes (modulus) [**2048 bits, 3072 bits, 4096 bits**]
- Elliptic Curve Digital Signature Algorithm and cryptographic key sizes [**256 bits, 384 bits, 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*, *P-521*]; ISO/IEC 14888-3, Section 6.4

].

4.2.2.6 Cryptographic Operation (Hash Algorithm) (FCS_COP.1/Hash)

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 message digest sizes [160, 256, 384, 512] bits that meet the following: ISO/IEC 10118-3:2004.

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

FCS_COP.1.1/KeyedHash 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 bits**] and message digest sizes [160, 256, 384, 512] bits that meet the following: ISO/IEC 9797-2:2011, Section 7 “MAC Algorithm 2”.

4.2.2.8 HTTPS Protocol (FCS_HTTPS_EXT.1)

FCS_HTTPS_EXT.1.1 The TSF shall implement the HTTPS protocol that complies with RFC 2818.

FCS_HTTPS_EXT.1.2 The TSF shall implement HTTPS using TLS.

FCS_HTTPS_EXT.1.3 If a peer certificate is presented, the TSF shall [not require client authentication] if the peer certificate is deemed invalid.

Application Note: *FCS_HTTPS_EXT.1.3 is only applicable in the case where the TOE supports mutual authentication or acts as an HTTPS Client. The TOE does not support mutual authentication and only implements HTTPS as a server. Therefore, peer certificates are not requested and not checked by the TSF.*

4.2.2.9 NTP Protocol (FCS_NTP_EXT.1)

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 [*SHA1*, *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.

4.2.2.10 Random Bit Generation (FCS_RBG_EXT.1)

FCS_RBG_EXT.1.1 The TSF shall perform all deterministic random bit generation services in accordance with ISO/IEC 18031:2011 using [CTR_DRBG (AES)].

FCS_RBG_EXT.1.2 The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [[**1**] platform-based noise sources] 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.

4.2.2.11 SSH Server Protocol (FCS_SSHS_EXT.1)

FCS_SSHS_EXT.1.1 The TSF shall implement the SSH protocol in accordance with: RFCs 4251, 4252, 4253, 4254, [4256, 4344, 5656, 6668].

FCS_SSHS_EXT.1.2[3] The TSF shall ensure that the SSH protocol implementation supports the following user authentication methods as described in RFC 4252: public key-based, [password-based].

FCS_SSHS_EXT.1.3 The TSF shall ensure that, as described in RFC 4253, packets greater than [**256,000**] 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-ctr, aes256-ctr, aes128-gcm@openssh.com, aes256-gcm@openssh.com].

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

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

[3] Modified by TD0631 to clarify requirements for public key authentication for SSH server.

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 each encryption key is used to protect no more than one gigabyte of data. After any of the thresholds are reached, a rekey needs to be performed.

4.2.2.12 TLS Client Protocol without Mutual Authentication (FCS_TLSC_EXT.1)

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

- TLS_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_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_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

].

FCS_TLSC_EXT.1.2 The TSF shall verify that the presented identifier matches [the reference identifier per RFC 6125 section 6].

FCS_TLSC_EXT.1.3 When establishing a trusted channel, by default the TSF shall not establish a trusted channel if the server certificate is invalid. The TSF shall also [

- Not implement any administrator override mechanism

].

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

4.2.2.13 TLS Server Protocol without Mutual Authentication (FCS_TLSS_EXT.1)

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

- TLS_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_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_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

].

FCS_TLSS_EXT.1.2 The TSF shall deny connections from clients requesting SSL 2.0, SSL 3.0, TLS 1.0 and [TLS 1.1].

FCS_TLSS_EXT.1.3 The TSF shall perform key establishment for TLS using [Diffie-Hellman parameters with size [*2048 bits*], ECDHE curves [*secp256r1*, *secp384r1*, *secp521r1*] and no other curves].

FCS_TLSS_EXT.1.4 The TSF shall support [no session resumption or session tickets].

4.2.3 Identification and Authentication (FIA)

4.2.3.1 Authentication Failure Management (FIA_AFL.1)

FIA_AFL.1.1 The TSF shall detect when an Administrator configurable positive integer within [**1 and 65534**] 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 [**unlock**] is taken by an Administrator; prevent the offending Administrator from successfully establishing remote session using any authentication method that involves a password until an Administrator defined time period has elapsed].

4.2.3.2 Password Management (FIA_PMG_EXT.1)

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: [“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, “)”, [**“_”, “-“, “+”, “=”, “[“, “{“, “]”, “}”, “,”, “<”, “>”, “.”, “/”, “?”, “~”, “`”, “ “ (i.e. space separators)**]];

b) Minimum password length shall be configurable to between [**6**] and [**128**] characters.

4.2.3.3 Password-Based Authentication Mechanism (FIA_UAU_EXT.2)

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

4.2.3.4 Protected Authentication Feedback (FIA_UAU.7)

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.

4.2.3.5 User Identification and Authentication (FIA_UIA_EXT.1)

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.

4.2.3.6 X.509 Certificate Validation (FIA_X509_EXT.1/Rev)

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

- RFC 5280 certificate validation and certificate path validation supporting a minimum path length of three certificates.
- The certification path must terminate with a trusted CA certificate designated as a trust anchor.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of the certificate using [the Online Certificate Status Protocol (OCSP) as specified in RFC 6960].

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

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

4.2.3.7 X.509 Certificate Authentication (FIA_X509_EXT.2)

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

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

4.2.3.8 X.509 Certificate Requests (FIA_X509_EXT.3)

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.

4.2.4 Security Management (FMT)

4.2.4.1 Management of Security Functions Behavior (FMT_MOF.1/ManualUpdate)

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

4.2.4.2 Management of Security Functions Behavior (FMT_MOF.1/Services)

FMT_MOF.1.1/Services The TSF shall restrict the ability to start and stop services to Security Administrators.

4.2.4.3 Management of TSF Data (FMT_MTD.1/CoreData)

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

4.2.4.4 Management of TSF Data (FMT_MTD.1/CryptoKeys)

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

4.2.4.5 Specification of Management Functions (FMT_SMF.1)

FMT_SMF.1.1[4] The TSF shall be capable of performing the following management functions:

- Ability to administer the TOE locally and remotely;
- Ability to configure the access banner;
- Ability to configure the session inactivity time before session termination or locking;
- Ability to update the TOE, and to verify the updates using [digital signature] capability prior to installing those updates;
- Ability to configure the authentication failure parameters for FIA_AFL.1;
- [
    - Ability to start and stop services;
    - Ability to modify the behaviour of the transmission of audit data to an external IT entity;
    - Ability to manage the cryptographic keys;
    - Ability to re-enable an Administrator account;
    - Ability to set the time which is used for time-stamps;
    - Ability to configure NTP;
    - Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors;
    - Ability to import X.509v3 certificates to the TOE's trust store;
    - Ability to manage the trusted public keys database;

].

4.2.4.6 Restrictions on Security Roles (FMT_SMR.2)

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.

4.2.5 Protection of the TSF (FPT)

4.2.5.1 Protection of Administrator Passwords (FPT_APW_EXT.1)

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.

[4] Modified by TD0631 to clarify requirements for public key authentication for SSH server.

4.2.5.2 Protection of TSF Data (for reading of all pre-shared keys, symmetric keys, and private keys) (FPT_SKP_EXT.1)

FPT_SKP_EXT.1.1 The TSF shall prevent reading of all pre-shared keys, symmetric keys, and private keys.

4.2.5.3 Reliable Time Stamps (FPT_STM_EXT.1)

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

FPT_STM_EXT.1.2[5] The TSF shall [allow the Security Administrator to set the time, synchronise time with an NTP server].

4.2.5.4 TSF Testing (FPT_TST_EXT.1)

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: [**cryptographic algorithm Known Answer Tests (KATs), firmware/software integrity test**].

4.2.5.5 Trusted Update (FPT_TUD_EXT.1)

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.

4.2.6 TOE Access (FTA)

4.2.6.1 TSF-Initiated Session Locking (FTA_SSL_EXT.1)

FTA_SSL_EXT.1.1 The TSF shall, for local interactive sessions, [terminate the session] after a Security Administrator-specified time period of inactivity.

4.2.6.2 TSF-Initiated Termination (FTA_SSL.3)

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

4.2.6.3 User-Initiated Termination (FTA_SSL.4)

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

4.2.6.4 Default TOE Access Banners (FTA_TAB.1)

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

[5] Modified by TD0632 for consistency with time data for vNDs.

4.2.7 Trusted Path/Channels (FTP)

4.2.7.1 Inter-TSF Trusted Channel (FTP_ITC.1)

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

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

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for [**export of audit records to external syslog server**].

4.2.7.2 Trusted Path (FTP_TRP.1/Admin)

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

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

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

4.3 TOE Security Assurance Requirements

The security assurance requirements for the TOE are included by reference from the [cPPND].

Table 7: Assurance Components

Requirement Class	Requirement Component
ADV: Development	ADV_FSP.1 Basic functional specification
AGD: Guidance documents	AGD_OPE.1 Operational user guidance
	AGD_PRE.1 Preparative procedures
ALC: Life-cycle support	ALC_CMC.1 Labelling of the TOE
	ALC_CMS.1 TOE CM coverage
ATE: Tests	ATE_IND.1 Independent testing – conformance
AVA: Vulnerability assessment	AVA_VAN.1 Vulnerability survey

5 TOE Summary Specification

This section describes the following security functions implemented by the TOE to satisfy the SFRs claimed in Section 4.2:

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

5.1 Security Audit

5.1.1 Audit Data Generation

The TOE generates audit records for start-up and shut-down of the audit functions and the following administrative actions:

- Administrative login and logout.
- Changes to TSF data related to configuration changes (in addition to the information that a change occurred, the TOE logs 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 is logged).
- Resetting passwords (name of related user account is logged).

Additionally, the TOE logs the specifically defined auditable events listed in Table 6.

The TOE records the following information within each audit record: date and time of the event; type of event; subject identity; the outcome (success or failure) of the event; and for each audit event type, based on the auditable event definitions of the functional components included in the cPP/ST, information specified in column three of Table 6.

All auditable events that involve configuration of the TSF include the action requested, the success or failure, and the identity of the user that made the request. This includes any cryptographic operations, specifically requesting to generate a CSR, import of a certificate, management of certificate stores, and management of the list of certificates used to validate servers. All actions by administrators that involve keys are performed in association with certificates. When auditing certificate operations, the TOE records the Issuer and serial number field of a certificate in order to uniquely identify the certificate.

This aspect of the Security Audit security function satisfies FAU_GEN.1 and FAU_GEN.2.

5.1.2 Audit Storage and Audit Record Export

The TOE is a single standalone appliance that stores audit logs locally in the logs file (`/data/Audit`) and provides the administrator the ability to configure the real-time export of syslog records protected with TLS. To enable sending to a remote Syslog server, use the `REMOTESYSLOG` command.

The audit service rotates to a new file when the current file exceeds 5K in size. The audit log service maintains only the most recent 20 files, so local audit log space is capped at 100K.

The TOE does not provide any interfaces to modify or manually delete the stored audit logs and can only be viewed by a Security Administrator.

This aspect of the Security Audit security function satisfies FAU_STG.1 and FAU_STG_EXT.1.

5.2 Cryptographic Support

The TOE includes the Crestron Crypto Kernel for Open SSL libraries with OpenSSL 1.0.2zd that has obtained CAVP certificates for their cryptographic algorithms. Table 8 below summarizes the CAVP certificates. The TOE uses this library for the following functions: in support of its NTP implementation; to verify TOE update package signatures; all SSH, TLS and certificate functionality.

Table 8: Cryptographic Functions Implemented by OpenSSL

Functions	Standards	Certificates
Asymmetric Key Generation (FCS_CKM.1)		
RSA (2048 bits, 3072 bits, 4096 bits)	FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.3	RSA #A1221
ECDSA (P-256, P-384, P-521 curves)	FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.4	ECDSA #A1222
DSA (2048 bits)	FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.1	DSA #A1222
FFC Schemes using ‘safe-prime’ groups and RFC 3526	NIST Special Publication 800-56A Revision 3	CCTL Tested
Key establishment (FCS_CKM.2)		
Elliptic curve-based scheme	NIST Special Publication 800-56A Revision 2, “Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography”	KAS-ECC #A1223
Finite field-based scheme	NIST Special Publication 800-56A Revision 2, “Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography”	KAS-FFC #A1223
FFC Schemes using “safe-prime” groups	NIST Special Publication 800-56A Revision 3 and RFC 3526 groups	CCTL Tested
Data encryption (FCS_COP.1/DataEncryption)		
AES in CBC mode (128, 256 bits) AES in GCM mode (128, 256 bits) AES in CTR mode (128, 256 bits)	ISO 18033-3 (AES) ISO 10116 (CBC mode) ISO 10116 (CTR mode) ISO 19772 (GCM mode)	AES #A1222
Digital signature generation and verification (FCS_COP.1/SigGen)		
RSA Digital Signature Algorithm (2048 bit, 3072 bit, and 4096 bit modulus)	FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSAPKCS1v1_5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3	RSA #A1222

Functions	Standards	Certificates
ECDSA Elliptic Curve Digital Signature Algorithm (P-256, P-384, P-521 curves)	FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 6 and Appendix D, Implementing "NIST curves" [P-256, P-384, P-521]; ISO/IEC 14888-3, Section 6.4	ECDSA #A1222
Cryptographic hashing (FCS_COP.1/Hash)		
SHA-1 (digest size 160 bits) SHA-256 (digest size 256 bits) SHA-384 (digest size 384 bits) SHA-512 (digest size 512 bits)	ISO/IEC 10118-3:2004	SHS #A1222
Keyed-hash message authentication (FCS_COP.1/KeyedHash)		
HMAC-SHA-1 (key size 160 bits, digest size 160 bits) HMAC-SHA-256 (key size 256 bits, digest size 256 bits) HMAC-SHA-384 (key size 384 bits, digest size 384 bits) HMAC-SHA-512 (key size 512 bits, digest size 512 bits)	ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2"	HMAC #A1222
Deterministic random bit generation (FCS_RBG_EXT.1)		
AES-256 CTR_DRBG	ISO/IEC 18031:2011	DRBG #A1222

5.2.1 Cryptographic Operations

The TOE uses most cryptographic algorithms in support of TLS and SSH. For example, the TOE uses SHA hashing during digital signature calculation (hashing of the message); for integrity as part of HMAC-SHA operations within TLS; and also for authentication of NTP servers. The TOE implements AES-CBC and GCM (both 128 and 256-bit) depending upon the TLS cipher suite; and AES-CTR (both 128 and 256-bit) for SSH.

The TOE will generate ephemeral 2048-bit Diffie-Hellman (i.e., finite field) parameters or ECDSA P-256/P-384/P-521 ephemeral elliptic curve parameters for key establishment in TLS/SSH. The TOE will use 2048/3072/4096 RSA or ECDSA (P-256/P-384/P-521) signature generation and verification operations as part of server authentication, depending on whether the negotiated cipher suite specifies the use of an RSA-based or ECDSA-based X.509 certificate. Additionally, RSA 2048 bit signatures are used when verifying TOE update packages.

The TOE performs SHA-1/256/384/512 cryptographic hashing with message digest sizes 160, 256, 384, 512, in bits that meets ISO/IEC 10118-3:2004. The TOE performs HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, or HMAC-SHA-512 (depending on negotiated cipher suite) for integrity of TLS protected data using SHA-1/256/384/512 with key sizes of 160/256/384/512 bits to produce a 160/256/384/512 output MAC. The HMAC-SHA-1 and HMAC-SHA-256 algorithms have a block size of 512 bits, while the HMAC-SHA-384 and HMAC-SHA-512 algorithms have a block size of 1024 bits. The keyed-hash message authentication cryptographic algorithms have the following cryptographic key sizes: for HMAC-SHA-1, 160 bits; for HMAC-SHA2-256, 256 bits; for HMAC-SHA2-384, 384 bits; and for HMAC-SHA2-512, 512 bits.

This aspect of the Cryptographic Support security function satisfies FCS_COP.1/DataEncryption, FCS_COP.1/SigGen, FCS_COP.1/Hash, and FCS_COP.1/KeyedHash.

5.2.2 Random Bit Generation

The TOE uses an AES-256 CTR_DRBG from the OpenSSL library. The maximum security strength supported by the DRBG is the security strength of the block cipher (256 bits). The TOE instantiates the DRBG with the maximum security strength, obtaining the 256 bit seed (along with a 128 bit nonce) by calling `/dev/random` for 384 bits. The TOE accumulates entropy into `/dev/random` using the `jitterentropy-rngd` package (version 1.2.3), which collects entropy from CPU execution time jitter. The TOE assumes the output from the `jitterentropy-rngd` package provides 8 bits of entropy per byte (i.e., full entropy) when mixed into the entropy pool that feeds `/dev/random`. Since calls to `/dev/random` block until sufficient entropy is accumulated to satisfy the number of bits requested, the TOE will seed the CTR_DRBG with 256 bits of entropy.

The TOE uses this DRBG to generate all keys (as part of SSH, TLS and CSR key generation) as well as to generate salts and nonces (for password hashing and TLS respectively).

This aspect of the Cryptographic Support security function satisfies FCS_RBG_EXT.1.

5.2.3 Cryptographic Key Generation and Establishment

The TOE supports generating key pairs, both for authentication and for key exchange for SSH and TLS. When generating authentication key pairs, the TOE can generate a CSR with RSA 2048, 3072 and 4096 bit key pairs; and ECDSA P-256, P-384, P-521 key pairs.

For key establishment, the TOE will generate 2048-bit DHE keys or P-256/P-384/P-521 ECDHE keys (depending on the negotiated cipher suite) during TLS negotiation. For DHE, the TOE generates FIPS 186-type domain parameters conforming to parameter-size set FC as specified in NIST SP 800-56A Rev. 3, with *p* = 2048 bits and *q* = 256 bits.

The TOE performs key establishment for TLS, and SSH. The TOE acts as both a TLS server to service incoming administrative sessions and as a TLS client for syslog export. The TOE supports elliptic curve, and finite field-based key establishment for TLS, depending on the negotiated cipher suite. The TOE’s SSH implementation, used for administrative sessions, uses only diffie-hellman-group14-sha1 (finite field-based using “safe-prime” groups key establishment), ecdh-sha2-nistp256, ecdh-sha2-nistp384, and ecdh-sha2-nistp521 for its key exchange methods.

This aspect of the Cryptographic Support security function satisfies FCS_CKM.1 and FCS_CKM.2.

5.2.4 Cryptographic Key Destruction

The TOE clears keys from volatile memory by overwriting the memory locations in RAM with zeroes. The TOE clears plaintext keys in non-volatile storage by performing a single overwrite consisting of a new value of the key. Keys stored in flash are AES-256-CBC encrypted.

Table 9: Key Clearing

Key	Storage Location	How Stored	Usage and destruction
SSH Host Private Key	Flash	plaintext Root access only	Used by SSH for client authentication of the SSH server, which is the TOE. It is replaced with a new key at request of the administrator using the command: `SSHSERVER GENHOSTKEY`.
SSH Session Key	RAM	plaintext	Used by SSH to encrypt a session and is destroyed when the SSH session is closed.
TLS Session Key	RAM	plaintext	Used by TLS to encrypt a session and is destroyed when the TLS session is closed.
Diffie-Hellman Shared Secret	RAM	plaintext	Used by the SSH and TLS protocols and is destroyed at the completion of the DH exchange step of those protocols
Web Server Certificate Private Key	Flash & RAM	Encrypted, AES-256-CBC	Used for web server. Uploaded by user, can be deleted or replaced by user.

The TOE incorporates OpenSSL, which provides implementation of the cryptographic algorithms specified in Table 8. The TOE invokes the OpenSSL cryptomodule APIs to set up and maintain the full TLS session, using the underlying cryptographic algorithms as identified in Table 8. Therefore, all key generation, negotiation of session keys, and packet authentication is performed by the OpenSSL cryptomodule. Files such as private keys and certificates are manually uploaded to the TOE during initial setup. Changes can be performed by remote access (SSH or GUI) or locally on the appliance with physical access to a network port. The SSH Host Private Key is stored in a file in a filesystem that is located in Flash. When an administrator requests a new key, the command `SSHSERVER GENHOSTKEY` invokes lower-level file system APIs to open and write to the file.

The plaintext private keys and CSPs (see Table 9 above) are managed by the cryptomodule and stored in RAM or Flash. Encrypted flash keys are copied unencrypted to a RAM drive at startup for runtime use. The cryptomodule does not store any other secret or private keys or CSPs persistently (beyond the lifetime of an API call). All secret keys, plaintext private keys and CSPs are destroyed automatically by the API when no longer required by overwriting once with zeroes or with a new value of the key (for SSH private keys).

The Web Server certificate private keys are stored in a PKCS#12 file (in flash) with a randomly generated password. That password is stored in the configuration file located in the `/etc` directory encrypted with a password that is embedded in the firmware source code. The key and the password are stored encrypted using AES-256-CBC.

All keys, key material, and authentication credentials are protected from unauthorized disclosure. There are no configurations or circumstances that do not conform to the key destruction requirement.

This aspect of the Cryptographic Support security function satisfies FCS_CKM.4.

5.2.5 Cryptographic Protocols

The TOE implements the following cryptographic protocols to protect communications between itself and non-TOE entities:

- TLS as a client—the TOE acts as a TLS client when exporting audit records to an external audit server.
- TLS as a server—the TOE acts as a TLS server supporting inbound administrative sessions.

Mutual authentication is not supported for TLS.

- SSH—the TOE acts as an SSH server supporting inbound administrative sessions.
- NTP—the TOE can synchronize its system clock with an NTP server.

5.2.5.1 SSH Server Protocol

The TOE's SSH server implements the SSH protocol in accordance with RFCs 4251, 4252, 4253, 4254, 4256, 4344, 5656, 6668. No optional characteristics are supported. The TOE supports SSH public key-based and password-based authentication methods as described in RFC 4252 and as described in RFC 4253, packets greater than 256 kilobytes in an SSH transport connection are dropped.

For its SSH transport implementation the TOE uses the following encryption algorithms and rejects all other encryption algorithms: aes128-ctr, aes256-ctr, aes128-gcm@openssh.com, and aes256-gcm@openssh.com.

The SSH public-key based authentication implementation uses ssh-rsa, and ecdsa-sha2-nistp256 as its public key algorithms and rejects all other public key algorithms. The TOE supports these algorithms, and only these algorithms, both to authenticate clients attempting to connect to the TOE using public key-based authentication, and for the TOE's server host key algorithms used to authenticate the TOE to an SSH client.

The SSH transport implementation uses only hmac-sha2-256, hmac-sha2-512, and implicit GCM as its data integrity MAC algorithms and rejects all other MAC algorithms. Specifically, the TOE uses implicit GCM MACs for aes256-gcm@openssh.com and aes128-gcm@openssh.com.

The TOE's SSH implementation uses only diffie-hellman-group14-sha1, ecdh-sha2-nistp256, ecdh-sha2-nistp384, and ecdh-sha2-nistp521 for its key exchange methods.

Within SSH connections the same session keys are used for a threshold of no longer than one hour, and each encryption key is used to protect no more than one gigabyte of data. After any of the thresholds are reached, the TOE performs a rekey.

This aspect of the Cryptographic Support security function satisfies FCS_SSHS_EXT.1

5.2.5.2 TLS Client Protocol

The TOE's TLS client supports TLS 1.2 with the following TLS 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_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_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_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

The TOE allows the TLS client (used for audit export) to specify the syslog server by hostname (which uses the internal `hosts` file for resolution). Certificate validation is done for validity (e.g., cryptographically valid, not expired, not revoked). The TOE's TLS client implementation establishes its reference identifiers from the administrator-configured reference identifiers per Section 6 of RFC 6125, using the hostname as a reference identifier and checking that the syslog server's certificate includes the specified identifier. The FQDN in the CN of the certificate is validated against the expected value per RFC 6125 section 6, and the certificate is rejected if the reference identifier is invalid. The TLS client supports the Elliptic Curves Extension (specifying only secp256r1, secp384r1, and secp521r1) in its Client Hello, and the TOE does not require (nor allow) administrative configuration of this extension; the TOE always sends it. Wildcards are supported.

The TOE's TLS client does not support mutual authentication or IP addresses.

This aspect of the Cryptographic Support security function satisfies FCS_TLSC_EXT.1.

5.2.5.3 TLS Server/HTTPS Protocols

The TOE's TLS server supports TLS 1.2 with the same TLS ciphersuites as identified above for the TOE's TLS Client.

The TOE denies versions of TLS older than TLS 1.2 and does not support session resumption or session tickets. The TOE configures its OpenSSL instance to reject older versions of TLS by setting the SSL_CTX_set_min_proto_version parameter to the value corresponding to TLS 1.2, as seen in the following code snippet:

```
if (SSL_CTX_set_min_proto_version(pEth->ctx, TLS1_2_VERSION) == 0)

       ErrorPrintf(eERROR_LEVEL_WARNING, "SSL_CTX_set_min_proto_version

for TLS1_2_VERSION returned error\n");

".
```

For TLS key establishment, the TOE supports 2048-bit DHE as described in section 5.2.3 above and the following ECDHE curves: secp256r1; secp384r1 and secp521r1. The TLS Server doesn't do certificate validation. It sends its certificate to the external TLS client so that the TLS client can validate the TLS server's certificate.

The TOE implements TLS server functionality for inbound management requests over HTTPS. The TOE's HTTPS implementation conforms to RFC 2818. Since it is used only for Web UI management connections, only the server requirements in RFC 2818 are applicable to the TOE. RFC 2818 (HTTP Over TLS) describes how to use TLS to secure HTTP connections over the Internet. The details of the TOE behavior with regards to the following requirements listed in RFC 2818 are as follows:

- Section 2.1, Connection Initiation:
    - The TOE acts as the HTTPS Server, therefore this section does not pertain to the TOE
- Section 2.2, Connection Closure:
    - The TOE sends TLS closure alert when terminating an HTTPS connection
    - The TOE does not support session reuse
    - The TOE meets described behavior; no deviation
- Section 2.2.1, Client Behavior:
    - The TOE acts as the HTTPS Server, therefore this section does not pertain to the TOE
- Section 2.2.2, Server Behavior:
    - The TOE does not support session resumption
    - The TOE will attempt to initiate an exchange of closure alerts with the client before closing the connection
- Section 2.3, Port Number:
    - The TOE utilizes TCP port 443 to listen for incoming HTTPS connections
- Section 2.4, URI Format:
    - The TOE supports and requires the "https://" URI protocol identifier prefix for incoming HTTPS requests
- Section 3.1, Server Identity:
    - Pertains to client behavior, not applicable to the TOE
- Section 3.2, Client Identity:
    - The TOE does not support mutual authentication with regards to the HTTPS interface.

This aspect of the Cryptographic Support security function satisfies FCS_HTTPS_EXT.1, and FCS_TLSS_EXT.1.

5.2.5.4 NTP Protocol

The TOE can synchronize its system clock with up to 3 NTP servers. The TOE supports NTP v4 as defined in RFC 5905 and can use SHA-1 or SHA-256 as its means for authenticating the NTP timestamps it receives from configured NTP servers. The TOE will not update NTP timestamps from broadcast or multicast addresses.

This aspect of the Cryptographic Support security function satisfies FCS_NTP_EXT.1.

5.3 Identification and Authentication

5.3.1 User Identification and Authentication

During initial configuration, the TOE's default admin password must be changed to conform to the password policy requirements. The default admin can define additional users. Username/password credentials (or key for SSH) are necessary to log in and access the management functions. Passwords or keys can be used for SSH connections (local via direct connection to a network port, or remote). The remote GUI access only allows password-based connection.

When a password is used, the TOE first checks the validity of the authentication against the accounts defined locally within the TOE. If an account with the asserted identity is not found, the user login is rejected. For a successful login, the identity and password must match those defined in the local account. A key presented for SSH login must match that for the defined account in the TOE's database. If the asserted identity and password or key cannot be verified then the login fails and an audit record is generated.

The TOE does not allow any security-relevant actions prior to requiring the identification and authentication process except the display of the warning banner. For the GUI the user must acknowledge the banner by clicking on an '**OK**' button. The TOE provides no feedback to the administrator during authentication other than the success or failure of their attempt. For local password-based credentials, the TOE accepts passwords that consist of any combination of uppercase letters, lowercase letters, numbers, and special characters. The TOE supports characters defined by the following regular expression: ^[\p{L}\p{N}\p{Zs}\p{S}\p{P}]*$. This includes all letters, numbers, symbols, punctuation, and space separators. The `SETPASSWORDRULE` command is used to change the minimum password length to values between 6 and 128 characters (default is 8 characters).

This aspect of the Identification and Authentication security function satisfies FIA_UIA_EXT.1, FIA_UAU_EXT.2, FIA_UAU.7, and FIA_PMG_EXT.1.

5.3.2 Authentication Failure Management

For password-based authentication, the administrator can configure the number of incorrect remote authentication attempts enforced on user accounts to a value between 1 and 65534 (with the default being 5). The TOE maintains a counter for each user account that it increments for each unsuccessful password-based authentication attempt. The counter is reset to zero on a successful authentication attempt. If an unsuccessful authentication attempt causes the counter to reach the configured threshold, the TOE will enforce an administrator-configurable (between 1 and 255 hours) lockout of the remote administrator. The admin can also unlock a user with the `REMLOCKEDUSER` command. The `SETUSERLOGINATTEMPTS` and `SETUSERLOCKOUTTIME` commands are used to configure the thresholds.

Authentication failure handling is only enforced on password credentials. User authentication based on SSH private key is not subject to the lockout mechanism. To ensure there is never a case where all administrators are locked out due to the session locking mechanism, at least one admin account must be set up to use SSH public key authentication.

This aspect of the Identification and Authentication security function satisfies FIA_AFL.1.

5.3.3 X.509 Certificate Validation

The TOE performs revocation checking of certificates during TLS client connection establishment (i.e., when the TOE acts as a TLS client connecting to a remote syslog-tls server) using the Online Certificate Status Protocol (OCSP) as specified in RFC 6960. OCSP certificates presented for OCSP responses have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.

The TOE performs the revocation checking after having checked the validity of the server certificate and its chain, conformant to RFC 5280. This includes supporting a minimum path length of three certificates and the certification path terminating with a trusted CA certificate designated as a trust anchor. The TOE requires that TLS server certificates have the Server Authentication purpose and that TLS client certificates have the Client Authentication purpose in order to be considered valid. The TOE will not treat a CA certificate as such unless the basicConstraints extension is present with the CA flag set to TRUE.

The TOE does not use certificates for trusted updates or executable code integrity verification and therefore the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) is not used in the extendedKeyUsage field.

This aspect of the Identification and Authentication security function satisfies FIA_X509_EXT.1/Rev.

5.3.4 X.509 Certificate Authentication

The TOE uses X.509 certificates for authentication of TLS and HTTPS trusted channels. The TOE relies upon the administrator to load the CA certificates (root CA and any needed intermediate certificates).

When receiving a connection from a remote administrator, the TSF will present its own server certificate to the client. Mutual authentication is not supported and there are no administrative over-rides.

During revocation checking of a presented server certificate, if the TOE’s TLS client cannot establish a connection to determine revocation status, the connection will not be established.

This aspect of the Identification and Authentication security function satisfies FIA_X509_EXT.2.

5.3.5 X.509 Certificate Requests

The TOE allows an administrator to request the TOE perform on-board key generation of an RSA key pair and then outputs a Certificate Signing Request (CSR), which the administrator can have signed by a suitable CA.

This aspect of the Identification and Authentication security function satisfies FIA_X509_EXT.3.

5.4 Security Management

5.4.1 Security Roles and Specification of Management Functions

The TOE provides a number of roles for local and remote management of the TOE, of which only the ‘Administrator’ role corresponds to the Security Administrator as defined in the PP. The unprivileged roles are:

- Programmer– unprivileged. Users with this role do not have privileges to manage the TOE.
- Operator– unprivileged. Allows the user to reboot and monitor the appliance only. Users with this role do not have privileges to manage the TOE.
- User– unprivileged. Users with this role do not have privileges to manage the TOE.
- Connect– unprivileged. Users with this role do not have privileges to manage the TOE.

The TOE provides the Security Administrator administrative access through its HTTPS server and via a CLI. The CLI can be accessed remotely over SSH or locally by directly connecting a user laptop to a network port and using SSH. The web-based interface is accessed remotely from a web browser. The TOE provides the following management functions:

- Configure the access banner – CLI only
- Configure the session inactivity time before session termination – CLI only
- Update the TOE and verify TOE updates prior to installation using digital signature verification – CLI, Web UI

- Configure the authentication failure parameters – CLI only
- Start and stop services (specifically, enable export of audit records to external audit server) – CLI only
- Modify the behaviour of the transmission of audit data to an external IT entity – CLI only
- Manage cryptographic keys: CSRs and TLS private key for web server – CLI only
- Ability to re-enable an Administrator account – CLI only
- Set the time used for time stamps – CLI, Web UI
- Configure NTP – CLI, Web UI
- Manage the TOE's trust store and designate X.509 v3 certificates as trust anchors – CLI only
- Import X.509 v3 certificates to the TOE's trust store (web server) – CLI only
- Manage trusted public keys for SSH client authentication – CLI only.

This aspect of the Security Management security function satisfies FMT_SMR.2 and FMT_SMF.1.

5.4.2 Management of Security Functions Behavior

The ability to perform TOE updates and to start and stop services (specifically, to enable and disable export of audit records to an external syslog server using the `REMOTESYSLOG` CLI command) is restricted to the single administrative role of Security Administrator using role-based access control.

This aspect of the Security Management security function satisfies FMT_MOF.1/ManualUpdate and FMT_MOF.1/Services.

5.4.3 Management of TSF Data

The TOE does not provide any access to management functions prior to authentication and restricts the ability to manage cryptographic keys to Security Administrators using role-based access control methods. Specifically, the Security Administrator may use the TOE to generate CSRs (which contain key pairs) and load certificates (whether it is certificate for the TOE generated by an external CA or a certificate used to validate a presented TLS client or server certificate) into the TOE's Trust Store. The Security Administrator may additionally cause the TOE to generate SSH Host keys to be used by the TOE and import SSH public keys to associate with TOE user accounts. Certificate management commands available to authorized administrators from the console are:

Table 10: Console Certificate Management Commands

Command	Description
`CREATECSR`	Generates a Certificate Signing Request that can be downloaded from the device.
`CERTIFICATE`	Manage device certificate stores.
`SSLVERIFY`	Enable/disable certificate validation options.
`SSHSERVER GENHOSTKEY`	Generates a new SSH host private key.
`ADDUSERKEY`	Associates an SSH public key to a user.

A certificate signing request generates an internal server key (for the web management interface) and Certificate signing request file. This will create a `request.csr` file in the `/SYS` folder of the NVX device.

This aspect of the Security Management security function satisfies FMT_MTD.1/CoreData and FMT_MTD.1/CryptoKeys.

5.5 Protection of the TSF

5.5.1 Protection of Administrator Passwords

The TOE stores administrative password data in a salted SHA-512 hash within an internal configuration file. The TOE does not provide interfaces for an administrator to view, extract, or read the password. The TOE only accepts passwords during authentication attempts (or during administrative changing of the password), salts and hashes the provided password (and then either compares it to the stored value or stores the new value depending upon whether the administrator is authenticating or changing the administrative password).

This aspect of the TSF Protection security function satisfies FPT_APW_EXT.1.

5.5.2 Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)

The TOE stores its Web Server Certificate private key internally in flash memory encrypted using AES-256-CBC and does not provide any commands to access them. The TOE stores SSH private host key in plaintext in flash and does not provide any interfaces to view the key. The private keys are used for the HTTPS and SSH management connections.

This aspect of the TSF Protection security function satisfies FPT_SKP_EXT.1.

5.5.3 TSF Testing

The TOE performs a series of power-up Known Answer Tests (a KAT for each library cryptographic algorithm that the TOE utilizes) as well as an integrity test during power-up. In particular, the TOE executes the following KATs: AES, RSA, ECDSA, DH, ECDH, DRBG, HMAC, and SHA. The self-test for verification of the integrity of the TOE firmware and software uses RSA 2048-bit signature and SHA-256 hash algorithm. For each self-test, the TOE uses known inputs to calculate an expected cryptographic result, and compares that result to the known result. If the calculated result matches the expected result, the test passes; if it does not match, the test fails. A failure of a power-up self-test causes the TOE to halt its boot and reboot.

The self-tests are sufficient to demonstrate that the TSF is operating correctly since they encompass the cryptographic functionality and the integrity of the entire TOE firmware executable code.

This aspect of the TSF Protection security function satisfies FPT_TST_EXT.1.

5.5.4 Trusted Update

The TOE provides an administrator the ability to view the currently executing version of firmware (for example, on the **STATUS** tab in the navigation bar of the web-based user interface, one can see the model name, serial number, and firmware version).

The TOE supports manual trusted updates for its firmware. Crestron digitally signs its firmware files using RSA 2048 bit and SHA-256.

The administrator can manually initiate updates to the TOE firmware through the GUI **Upload Firmware File** radio button. To initiate the firmware upgrade, the administrator must first obtain the firmware file by loading it onto the device and clicking **OK**. Specifically, the firmware file is available by entering: https://www.crestron.com/Support/Resource-Library into a web browser and entering "NVX" into the search box and then finding the TOE firmware version in the list. The administrator must enter their customer information and credentials to download the file prior to downloading the file to the `/firmware` location on the device. The TOE automatically verifies the digital signature on the update files during the download process and only installs the updates if the signatures verify. From the web interface, it is not possible to download an update and then install it at a later date.

Updates can also be manually initiated from the CLI. The update files must first be obtained by the administrator from Crestron's website as specified above but over SFTP. It is possible to upload a file and then install it at a later time. The file should be copied to the `/firmware` location on the device. The file is not active until it is installed by the administrator. There are no specific commands for viewing the version of downloaded but uninstalled firmware, however the version of the firmware is identified in the filename and can be queried by navigating to the firmware directory and examining the filename. Once installed, it is immediately activated. When the firmware update files are uploaded over SFTP, the files are verified at installation, not when the file is uploaded. If the digital signatures cannot be verified then the TOE will not perform the update and the failure will be audited.

In the evaluated configuration, the TOE does not provide an automated update mechanism.

When an upgrade fails the device remains in the current version and this can be verified by running the '`ver -v`' command. All failures are recorded in the syslog.

This aspect of the TSF Protection security function satisfies FPT_TUD_EXT.1.

5.5.5 Reliable Time Stamps

The TOE utilizes time when creating audit records and when checking syslog server and administrative client certificates (for expiration and revocation), for session inactivity timeout, and for administrator lock out periods. The TOE obtains and maintains time by allowing the administrator to both manually specify the time as well as configure up to 3 NTP servers with which the TOE synchronizes its clock. The TOE also has a battery-backed real-time clock that is used to guarantee the availability of time data.

This aspect of the TSF Protection security function satisfies FPT_STM_EXT.1.

5.6 TOE Access

5.6.1 Access Banner

The TOE provides the administrator the ability to set a banner that the TOE displays before each local and remote administrator login. Administrators login to the TOE remotely through the TOE's SSH or TLS-protected management ports, or locally via direct connection to a network port and using SSH. The single configured banner is displayed at all interfaces and can be configured by using an SFTP client to copy a file containing the text the administrator wishes to display onto the device into the /`SSHBanner/banner.txt` file.

This aspect of the TOE Access security function satisfies FTA_TAB.1.

5.6.2 Session Termination

The TOE terminates inactive sessions when the administrator configurable inactive timeout value is reached. The TOE provides two mechanisms, one for the CLI, which can be accessed using SSH, either remotely or locally by directly connecting a laptop to a network port, and a second for the Web UI, which is accessible only remotely via HTTPS. Session timeout for the CLI is configured with `SETLOGOFFIDLETIME` command. The default is 20 minutes and can be configured to values between 1 and 60 minutes. ‘0’ disables the inactive timeout feature. For the web server, the inactive time is configured with the `WEBSERVER TIMEOUT` command and can be configured for between 600 and 3600 seconds (1200 seconds is the default). The configuration will take effect during the next administrative session.

The Administrator can disconnect their administrative session from the CLI using the `BYE` command that will logout the user. From the web GUI, the administrator can click on the profile icon/image in the upper right and select **Sign Out**.

This aspect of the TOE Access security function satisfies FTA_SSL_EXT.1, FTA_SSL.3, and FTA_SSL.4.

5.7 Trusted Path/Channels

The TOE provides the administrator the ability to export audit records to a TLS-protected syslog server. The TOE’s TLS syslog configurations allows the administrator to specify the root CA certificate (as well as any needed intermediate CA certificates and any CRLs for revocation) for the TOE to use when validating the syslog server’s certificates.

The administrator can connect to the TOE using its HTTPS or SSH protected administration channel.

The Trusted Path/Channels security function satisfies FTP_ITC.1 and FTP_TRP.1/Admin.

6 Protection Profile Claims

This ST conforms to the collaborative Protection Profile for Network Devices, Version 2.2e, 23 March 2020 [cPPND] and including the following optional and selection-based SFRs: FAU_STG.1, FCS_HTTPS_EXT.1; FCS_NTP_EXT.1; FCS_SSHS_EXT.1; FCS_TLSC_EXT.1; FCS_TLSS_EXT.1; FIA_X509_EXT.1/Rev; FIA_X509_EXT.2; FIA_X509_EXT.3; FMT_MOF.1/Services; and FMT_MTD.1/CryptoKeys.As explained in Section 2, Security Problem Definition, the Security Problem Definition of the [cPPND] has been included by reference into this ST.

As explained in Section 3, Security Objectives, the ST reproduces the security objectives for the operational environment from [cPPND].

As explained in Section 4, IT Security Requirements, the SFRs have all been drawn from [cPPND]. As such, operations on SFRs already performed in that PP are not identified in this ST. Rather; the SFRs have been copied from [cPPND] and any formatting used in that PP has been removed. Operations performed on SFRs in the writing of this ST are identified in accordance with the conventions described in Section 1.3.

The SARs for the TOE are included by reference from the [cPPND].

7 Rationale

This ST includes by reference the [cPPND] Security Problem Definition and SARs and reproduces the security objectives for the Operational Environment. The ST makes no additions to the [cPPND] assumptions. [cPPND] SFRs have been reproduced with the Protection Profile operations completed. Operations on the SFRs follow [cPPND] application notes and assurance activities. Consequently, [cPPND] rationale applies but is incomplete. The TOE Summary Specification rationale below serves to complete the rationale required for the security target.

7.1 TOE Summary Specification Rationale

Each subsection in Section 5, the TOE Summary Specification (TSS), describes a security function of the TOE. Each description identifies the SFRs that are covered by that description and, as such, provides the rationale that indicates which requirements are satisfied by aspects of the corresponding security function. The security functions work together to satisfy all of the security functional requirements. Furthermore, all of the security functions are necessary in order for the TSF to provide the required security functionality.

This section, in conjunction with the TSS, provides evidence that the security functions are suitable to meet the TOE security requirements. The collection of security functions work together to provide all of the security requirements. The security functions described in the TSS are all necessary for the required security functionality in the TSF. Table 11: Security Functions vs. Requirements Mapping summarizes the relationship between security requirements and security functions.

Table 11: Security Functions vs. Requirements Mapping

	Security audit	Cryptographic support	Identification and authentication	Security management	Protection of the TSF	TOE access	Trusted path/channels
FAU_GEN.1	X						
FAU_GEN.2	X						
FAU_STG.1	X						
FAU_STG_EXT.1	X						
FCS_CKM.1		X					
FCS_CKM.2		X					
FCS_CKM.4		X					
FCS_COP.1/DataEncryption		X					
FCS_COP.1/SigGen		X					
FCS_COP.1/Hash		X					
FCS_COP.1/KeyedHash		X					
FCS_NTP_EXT.1		X					

	Security audit	Cryptographic support	Identification and authentication	Security management	Protection of the TSF	TOE access	Trusted path/channels
FCS_RBG_EXT.1		X					
FCS_SSHS_EXT.1		X					
FCS_TLSC_EXT.1		X					
FCS_TLSS_EXT.1		X					
FIA_AFL_EXT.1			X				
FIA_PMG_EXT.1			X				
FIA_UIA_EXT.1			X				
FIA_UAU_EXT.2			X				
FIA_UAU.7			X				
FIA_X509_EXT.1/Rev			X				
FIA_X509_EXT.2			X				
FIA_X509_EXT.3			X				
FMT_MOF.1/ManualUpdate				X			
FMT_MOF.1/Services				X			
FMT_MTD.1/CoreData				X			
FMT_SMF.1				X			
FMT_SMR.2				X			
FPT_APW_EXT.1					X		
FPT_SKP_EXT.1					X		
FPT_TST_EXT.1					X		
FPT_TUD_EXT.1					X		
FPT_STM_EXT.1					X		
FTA_SSL_EXT.1						X	
FTA_SSL.3						X	
FTA_SSL.4						X	
FTA_TAB.1						X	
FTP_ITC.1							X
FTP_TRP.1/Admin							X