Assurance Activity Report for
FireEye AX, CM, EX, FX, HX, NX, and VX Series Appliances running TRFEOS 10.0.4

FireEye AX, CM, EX, FX, HX, NX, and VX Series Appliances running TRFEOS 10.0.4
Security Target
Version 2.0

collaborative Protection Profile for Network Devices, Version 2.2e

AAR Version 1.7, 10/17/2024

Evaluated by:

2400 Research Blvd, Suite 395
Rockville, MD 20850

Prepared for:

National Information Assurance Partnership
Common Criteria Evaluation and Validation Scheme

The Developer of the TOE:
Trellix FireEye Security Holdings US LLC

The Author of the Security Target:
Acumen Security, LLC.

The TOE Evaluation was Sponsored by:
Trellix FireEye Security Holdings US LLC

Evaluation Personnel:
Reema Nagwekar
Saniya Shaikh
Yogesh Pawar

Common Criteria Version
Common Criteria Version 3.1 Revision 5

Common Evaluation Methodology Version
CEM Version 3.1 Revision 5

REVISION HISTORY

VERSION	DATE	CHANGES
1.0	06/12/2024	Initial Release
1.1	07/17/2024	Updated few sections
1.2	07/29/2024	Updated few sections
1.3	08/21/2024	Addressed ECR comments
1.4	08/28/2024	Updated references for ST document
1.5	08/28/2024	Updated CVE search date
1.6	10/04/2024	Updated CVE search date
1.7	10/17/2024	Updated software version to 10.0.4

CONTENTS

1 TOE OVERVIEW

FireEye AX, CM, EX, FX, HX, NX, and VX Series are network devices comprised of hardware and software. The virtual devices as defined in Table 1 [ST] are considered virtual network devices as defined in Case 1 of NDcPP 2.2e running on general purpose hardware and virtualization system which are outside of the TOE. In the virtual case, the TOE boundary represents the virtual network device only. The hardware appliances are physical devices comprised of the TOE firmware running on bare metal, where the TOE boundary is inclusive of hardware and software. The Trellix Appliances runs on a pre-installed, hardened TRFE(Trellix FireEye) operating system(TRFEOS) and comes pre-loaded with the TRFEOS software. TRFEOS runs on all platforms with version 10.0.4. Please see Section 1.1 for additional details on the TOE models.

The FireEye Malware Analysis (AX) series is a group of forensic analysis platforms that give security analysts hands-on control over powerful auto-configured test environments to safely execute and inspect advanced malware, zero-day and advanced persistent threat (APT) attacks embedded in Web pages, email attachments and files.

FireEye Central Management (CM) series consolidates the administration, reporting and data sharing of the FireEye products in one easy-to-deploy, network-based solution.

The FireEye Email Security (EX) Series Appliances are network devices that secure against advanced email attacks by using signature-less technology to analyze email attachments and quarantine malicious emails.

The FireEye Threat Prevention (FX) platform protects data assets against attacks originating in a wide range of file types. Web mail, online file transfer tools, the cloud, and portable file storage devices can introduce malware that can then spread to file shares and content repositories.

The FireEye Endpoint Security (HX) Appliances are network devices providing organizations with the ability to continuously monitor endpoints for advanced malware and indicators of compromise.

FireEye Network Security (NX) is an effective cyber threat protection solution that helps organizations minimize the risk of costly breaches by accurately detecting and immediately stopping advanced, targeted and other evasive attacks hiding in Internet traffic.

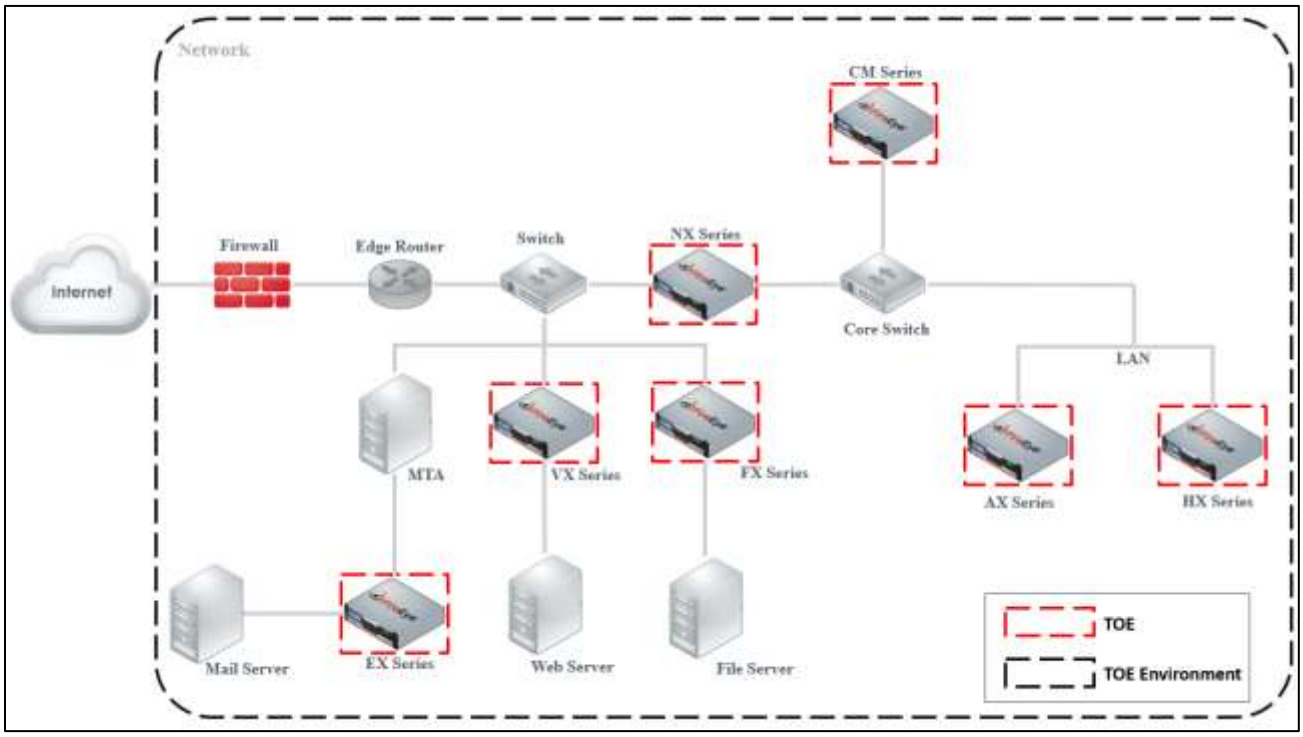
The FireEye Network Threat Prevention Platform (VX) identifies and blocks zero-day Web exploits, droppers (binaries), and multi-protocol callbacks to help organizations scale their advanced threat defenses across a range of deployments, from the multi-gigabit headquarters down to remote, branch, and mobile offices. FireEye Network with Intrusion Prevention System (IPS) technology further optimizes spend, substantially reduces false positives, and enables compliance while driving security across known and unknown threats.

Note: Each model of the TOE shares an identical codebase employing all NDcPP required functionality. Breach detection, email analysis, endpoint monitoring, IPS, malware analysis, and threat prevention

features are not evaluated as part of the Common Criteria certification and are excluded by the evaluation.

1.1 TOE DESCRIPTION

This section provides an overview of the TOE deployment, including physical boundaries, security functions, and relevant TOE documentation and references. Figure 1 below depicts a typical TOE deployment in a network. It provides a sample representation of where each of the FireEye AX, CM, EX, FX, HX, NX, and VX Series are typically deployed. The TOE is not distributed and does not require all variants or series to function. Instead, each model variant of each series is a standalone TOE. The purpose of Figure 1 is to represent how various instances of the TOEs are deployed in a typical network.



[1]**Figure 1 - Representative TOE Deployment**

1.1.1 PHYSICAL BOUNDARIES

Each instance of the TOE is a hardware and software solution implemented in one of the security appliance models listed in Table 1. The TOE guidance documentation that is considered to be part of the TOE can be found listed in the FireEye Common Criteria Addendum document and is downloadable from the FireEye website.

[1] Each instance of the TOE is a hardware and software solution implemented in one of the security appliance models and each of the different model is a standalone TOE.

The network on which the TOE resides is considered part of the environment. The software is pre-installed and is comprised of only the software versions identified above. In addition, software updates are downloadable from the FireEye website. A login ID and password is required to download the software update.

An instance of the TOE consists of a physical or virtual appliance instance of one of the models listed in Table 1.

Table 1 – TOE Physical Boundary Components

Model	CPU	Network Interfaces	Storage	Dimensions	Firmware
Physical Models					
AX5600	Intel Xeon E-2334 (Rocket Lake)	2x 1GigE BaseT	2 x 4TB disk / 4 TB virtual disk RAID 1	1 RU	TRFEOS 10.0.4
CM4600	Intel Xeon E-2334 (Rocket Lake)	2x 1GigE BaseT	4x 4TB disk / 8TB virtual disk RAID 10	1 RU	TRFEOS 10.0.4
CM7600	Intel Xeon Silver 4314 (Ice Lake)	2x 1GigE BaseT	4x 4TB disk / 8TB virtual disk RAID 10	2 RU	TRFEOS 10.0.4
CM9600	Intel Xeon Silver 4316 (Ice Lake)	2x 1GigE BaseT	4x 10TB disk / 20TB virtual disk RAID 10	2 RU	TRFEOS 10.0.4
EX3600	Intel Xeon E-2334 (Rocket Lake)	2x 1GigE BaseT	4x 4TB disk / 8TB virtual disk RAID 10	1 RU	TRFEOS 10.0.4
EX5600	Intel Xeon Silver 4314 (Ice Lake)	2x 1GigE BaseT	4x 4TB disk / 8TB virtual disk RAID 10	2 RU	TRFEOS 10.0.4
EX8600	Intel Xeon Silver 4316 (Ice Lake)	2x 1GigE BaseT	4x 4TB disk / 8TB virtual disk RAID 10	2 RU	TRFEOS 10.0.4
FX6600	Intel Xeon Silver 4316 (Ice Lake)	2x 1GigE BaseT	4x 4TB disk / 8TB virtual disk RAID 10	2 RU	TRFEOS 10.0.4
HX4600	Intel Xeon E-2378 (Rocket Lake)	2x 1GigE BaseT	4x 4TB disk / 8TB virtual disk RAID 10	1 RU	TRFEOS 10.0.4
NX2600	Intel Xeon E-2334 (Rocket Lake)	2x 1GigE BaseT	2 x 4TB disk / 4 TB virtual disk RAID 1	1 RU	TRFEOS 10.0.4

Model	CPU	Network Interfaces	Storage	Dimensions	Firmware
NX3600	Intel Xeon E-2378 (Rocket Lake)	2x 1GigE BaseT	2 x 4TB disk / 4 TB virtual disk RAID 1	1 RU	TRFEOS 10.0.4
NX4600	Intel Xeon Silver 4314 (Ice Lake)	2x 1GigE BaseT	2 x 4TB disk / 4 TB virtual disk RAID 1	2 RU	TRFEOS 10.0.4
NX5600	Intel Xeon Silver 4314 (Ice Lake)	2x 1GigE BaseT 2x 10G BaseT	2 x 4TB disk / 4 TB virtual disk RAID 1	2 RU	TRFEOS 10.0.4
NX6600	Intel Xeon Gold 6330 (Ice Lake)	2x 10G BaseT 2x SFP	2 x 10TB disk / 10TB virtual disk RAID 1	2 RU	TRFEOS 10.0.4
NX8600	Intel Xeon Platinum 8380 (Ice Lake)	2x 10G BaseT 2x SFP 2x 100G QSFP	2 x 10TB disk / 10TB virtual disk RAID 1	2 RU	TRFEOS 10.0.4
VX5600	Intel Xeon E-2334 (Rocket Lake)	2x 1GigE BaseT	2 x 4TB disk / 4 TB virtual disk RAID 1	1 RU	TRFEOS 10.0.4
VX12600	Intel Xeon Gold 6330 (Ice Lake)	2x 10G BaseT	4x 4TB disk / 8TB virtual disk RAID 10	2 RU	TRFEOS 10.0.4
Virtual Models					
CM7500V	ESXi Hypervisor v7.0 on Intel(R) Xeon(R) CPU E5-4620 v4 (Broadwell)	NA	NA	NA	TRFEOS 10.0.4
CM1500V	ESXi Hypervisor v7.0 on Intel(R) Xeon(R) CPU E5-4620 v4 (Broadwell)	NA	NA	NA	TRFEOS 10.0.4
CM2500V	ESXi Hypervisor v7.0 on Intel(R) Xeon(R) CPU E5-4620 v4 (Broadwell)	NA	NA	NA	TRFEOS 10.0.4
EX5500V	ESXi Hypervisor v7.0 on Intel(R) Xeon(R) CPU E5-4620 v4 (Broadwell)	NA	NA	NA	TRFEOS 10.0.4
FX2500V	ESXi Hypervisor v7.0 on Intel(R) Xeon(R) CPU E5-4620 v4 (Broadwell)	NA	NA	NA	TRFEOS 10.0.4

Model	CPU	Network Interfaces	Storage	Dimensions	Firmware
HX4502V	ESXi Hypervisor v7.0 on Intel(R) Xeon(R) CPU E5-4620 v4 (Broadwell)	NA	NA	NA	TRFEOS 10.0.4
HX4600V	ESXi Hypervisor v7.0 on Intel(R) Xeon(R) CPU E5-4620 v4 (Broadwell)	NA	NA	NA	TRFEOS 10.0.4
NX1500V	ESXi Hypervisor v7.0 on Intel(R) Xeon(R) CPU E5-4620 v4 (Broadwell)	NA	NA	NA	TRFEOS 10.0.4
NX2500V	ESXi Hypervisor v7.0 on Intel(R) Xeon(R) CPU E5-4620 v4 (Broadwell)	NA	NA	NA	TRFEOS 10.0.4
NX2550V	ESXi Hypervisor v7.0 on Intel(R) Xeon(R) CPU E5-4620 v4 (Broadwell)	NA	NA	NA	TRFEOS 10.0.4
NX4500V	ESXi Hypervisor v7.0 on Intel(R) Xeon(R) CPU E5-4620 v4 (Broadwell)	NA	NA	NA	TRFEOS 10.0.4
NX6500V	ESXi Hypervisor v7.0 on Intel(R) Xeon(R) CPU E5-4620 v4 (Broadwell)	NA	NA	NA	TRFEOS 10.0.4
NX7500V	ESXi Hypervisor v7.0 on Intel(R) Xeon(R) CPU E5-4620 v4 (Broadwell)	NA	NA	NA	TRFEOS 10.0.4
NX8500V	ESXi Hypervisor v7.0 on Intel(R) Xeon(R) CPU E5-4620 v4 (Broadwell)	NA	NA	NA	TRFEOS 10.0.4
NX10500V	ESXi Hypervisor v7.0 on Intel(R) Xeon(R) CPU E5-4620 v4 (Broadwell)	NA	NA	NA	TRFEOS 10.0.4

1.1.2 SECURITY FUNCTIONS PROVIDED BY THE TOE

The TOE provides the security functions required by the Collaborative Protection Profile for Network Devices, hereafter referred to as NDcPP v2.2e or NDcPP.

1.1.2.1 SECURITY AUDIT

The TOE keeps local and remote audit records of security relevant events. The TOE internally maintains the date and time which can be set manually or using authenticated NTP.

1.1.2.2 CRYPTOGRAPHIC SUPPORT

The TOE provides cryptographic support for the services described in Table 2. The related CAVP validation details are provided in Table 3.

Table 2 - TOE provided cryptography

Cryptographic Method	Use within the TOE
TLS Establishment	Used to establish initial TLS session
SSH Establishment	Used to establish initial SSH session
ECDSA Signature Services	Used in TLS session establishment
RSA Signature Services	Used in TLS session establishment Used in SSH session establishment Used in secure software update
Random Bit Generation	Used in TLS session establishment Used in SSH session establishment
Hashing	Used in secure software update Used in NTP integrity
HMAC	Used to provide TLS traffic integrity verification Used to provide SSH traffic integrity verification
AES	Used to encrypt TLS traffic Used to encrypt SSH traffic

The TOE utilizes Trellix OpenSSL FIPS Object Module cryptographic library.

For all cryptographic operations performed by the TOE, the cryptographic algorithms have been validated as identified in the table below.

Table 3 - CAVP Algorithm Testing References

Functions	Algorithms	Mode Supported	CAVP Certs.	Name	OE
Data Encryption	AES-CBC, AES-CTR, AES-GCM	CBC, CTR, GCM (128, 256)	A2624	Trellix OpenSSL FIPS Object Module	TRFEOS 10.0 on Intel(R) Xeon (R) E-2334 (Rocket Lake)
Hash	SHS (Cryptographic hashing)	SHA-1, SHA-256, SHA-384, SHA-512	A2624	Trellix OpenSSL FIPS Object Module	TRFEOS 10.0 on Intel(R) Xeon (R)

Random Number Generator	Counter DRBG HMAC DRBG	CTR_DRBG (AES-256), HMAC_DRBG(SHA-512)	A2624	Trellix OpenSSL FIPS Object Module	Gold 6330 (Ice Lake) TRFEOS 10.0 running on ESXi Hypervisor v7.0 on Intel(R) Xeon(R) CPU E5-4620 v4(Broadwell)
Key Generation	RSA KeyGen (FIPS186-4)	Mode: n(2048,3072), n = 2048,3072 SHA(256)	A2624	Trellix OpenSSL FIPS Object Module	
	ECDSA KeyGen (FIPS186-4) ECDSA KeyVer (FIPS186-4)	P-256, P-384, P-521	A2624	Trellix OpenSSL FIPS Object Module	
	DSA KeyGen (FIPS186-4)	(L,N): (2048,256)	A2624	Trellix OpenSSL FIPS Object Module	
	Safe Primes Key Generation	modp-2048(DH-14) modp-4096(DH-16) modp-8192(DH-18)	NA	No NIST CAVP, CCTL has performed all assurance/evaluation activities.	
Key Establishment	KAS ECC SSC Sp800-56Ar3 (Domain Parameter Generation)	P-256, P-384, P-521	A2624	Trellix OpenSSL FIPS Object Module	
	KAS-FFC-SSC Sp800-56Ar3 (safe-prime) (Domain Parameter Generation)	MODP-2048	A2624	Trellix OpenSSL FIPS Object Module	

	KAS-FFC-SSC Sp800-56Ar3 (Domain Parameter Generation)	modp-2048(DH-14) modp-4096(DH-16) modp-8192(DH-18)	NA	No NIST CAVP, CCTL has performed all assurance/evaluation activities.	
Digital Signature services	ECDSA SigGen (FIPS186-4) ECDSA SigVer (FIPS186-4)	P-256, P-384, P-521	A2624	Trellix OpenSSL FIPS Object Module	
	RSA SigGen (FIPS186-4) RSA SigVer (FIPS186-4)	Mode: n(2048, 3072), n = 2048,3072 SHA(256)	A2624	Trellix OpenSSL FIPS Object Module	
Keyed Hash	HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512	Mode: SHA-1, SHA-256, SHA-384, SHA-512	A2624	Trellix OpenSSL FIPS Object Module	

The Trellix OpenSSL FIPS Object Module provides cryptographic operations related to entropy.

1.1.2.3 IDENTIFICATION AND AUTHENTICATION

The TOE authenticates administrative users using a username/password combination. The TOE does not allow access to any administrative functions prior to successful authentication. The TOE validates and authenticates TLS clients and servers using X.509 certificates for all claimed certificate uses.

The TOE supports passwords consisting of alphanumeric and special characters and enforces minimum password lengths. The TSF supports authentication based on certificates. Certificates are used to authenticate trusted channels, not administrators. The TOE only allows users to view the login warning banner prior to authentication. Remote administrators are locked out after a configurable number of unsuccessful authentication attempts.

1.1.2.4 SECURITY MANAGEMENT

The TOE enables secure local and remote management of its security functions, including:

- Local console CLI administration

- Remote CLI administration via SSHv2
- Remote GUI administration via HTTPS/TLS[2]
- Administrator authentication using a local database
- Timed user lockout after multiple failed authentication attempts
- Password complexity enforcement
- Role Based Access Control - the TOE supports several types of administrative user roles. Collectively these roles comprise the "Security Administrator"
- Configurable banners to be displayed at login
- Timeouts to terminate administrative sessions after a set period of inactivity
- Protection of secret keys and passwords

1.1.2.5 PROTECTION OF THE TSF

The TOE ensures the authenticity and integrity of software updates through digital signatures and requires administrative intervention prior to the software updates being installed.

1.1.2.6 TOE ACCESS

The TOE can be configured to display a warning and consent banner when an administrator attempts to establish an interactive session over the CLI (local or remote) or remote web UI (Only VX series models don't support Web UI Feature). The TOE also enforces a configurable inactivity timeout for remote administrative sessions.

1.1.2.7 TRUSTED PATH/CHANNELS

The TOE protects the integrity and confidentiality of communications as follows:

- TLS connectivity with the following entities:
  - Audit Server
  - Management Web Browser[3]
- SSH connectivity with the following entities:
  - Management SSH Client

[2] VX series models doesn't support Web UI Feature and hence HTTPS and TLSS selection-based SFRs are not applicable to the VX Series Models
[3] VX series models doesn't support Web UI Feature and hence this selection-based SFR is not applicable to the VX Series Models

2 ASSURANCE ACTIVITIES IDENTIFICATION

The Assurance Activities contained within this document include all those defined within NDcPP v2.2e based upon the core SFRs and those implemented based on selections within the PPs/EPs.

3 TEST EQUIVALENCY JUSTIFICATION

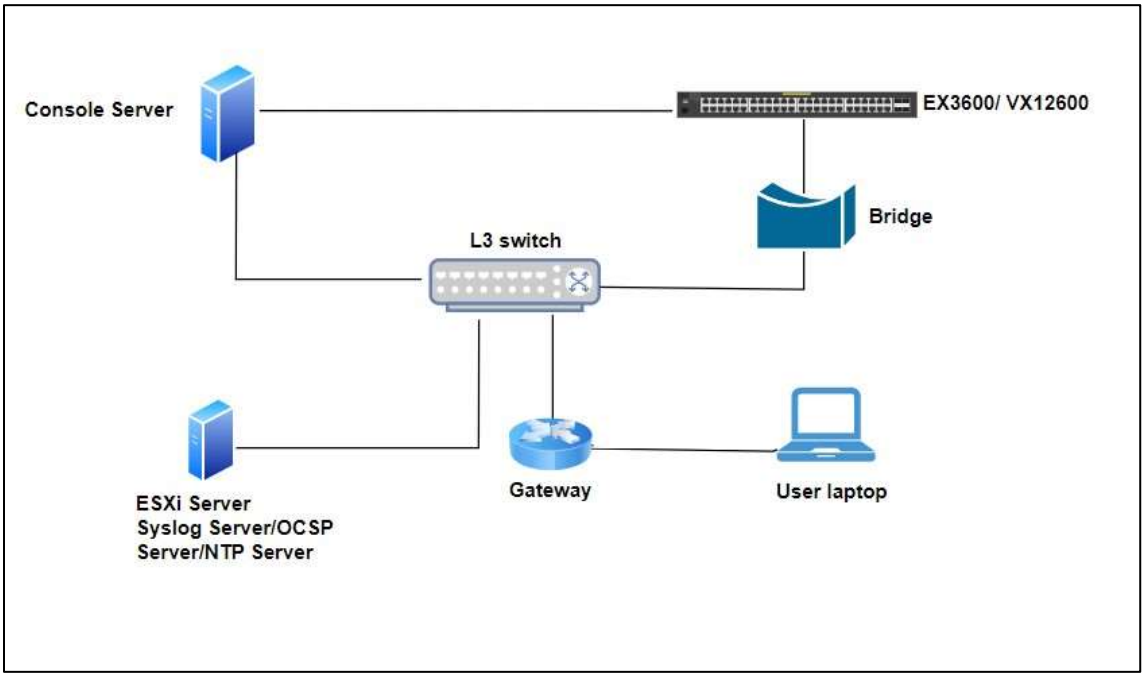
The equivalency analysis provides a per-category analysis of key areas of differentiation for each hardware model to determine the minimum subset used in the testing. The areas examined have used the areas and analysis description provided in the supporting documentation for the NDcPP. Additionally, a comparison of the data provided to identify a testing subset that will exercise each of the differences in TOE models. Based on the equivalency rationale, testing has been performed on the following subset:

- EX3600 running on TRFEOS 10.0.4 with Intel Xeon E-2334 (**Rocket Lake**)
- VX12600 running on TRFEOS 10.0.4 with Intel Xeon Gold 6330 (**Ice Lake**)
- CM2500V running on VMware vSphere ESXi 7.0 with TRFEOS 10.0.4 with Intel Xeon E5-4620 v4 (**Broadwell**).

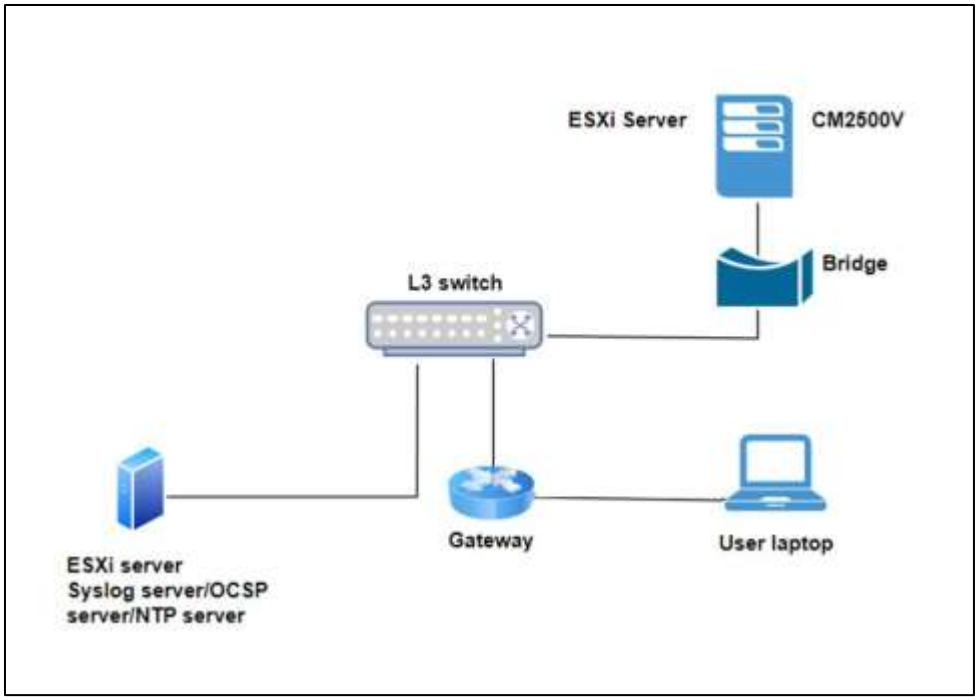
4 TEST BED DESCRIPTIONS

4.1 TEST BED

Physical TOE:



Virtual TOE:



4.2 CONFIGURATION INFORMATION

The following table provides configuration information about each device in the test environment.

Physical TOE:

Table 4 - Physical Device Details

Device Details Table								
Device Details		**Network Details**			**System Details**			
Device Name	**Function**	**IP Address**	**MAC Address**	**Protocols**	**OS, including version**	**Timing Source**	**Software & Tools, including version**	**Credentials**
EX3600 / VX12600	TOE	X.X.X.X	X:X:X:X:X:X	TLS/SSH	TRFEOS, 10.0.4	NTP Synced	N/A	N/A
User Laptop	Mgt. Access/Co	X.X.X.X	X:X:X:X:X:X	TLS/SSH	Windows 10	Manually set and verified	Chrome (Version 109.0.5414.120),	N/A

	nsole Access						Microsoft Edge (Version 110.0.1587.41), XCA (2.1.1) OpenSSL (1.1.1f) Putty (Release 0.77) Hex editor (Version 2.5.0.0)	
ESXi Server/ Syslog Server/ OCSP Server/NTP Server	ubuntuVM / Syslog server/ OCSP server TLS client TLS server NTP server SSH Client	X.X.X.X	X:X:X:X:X:X	TLS/SSH	Ubuntu 20.04.4	Manually set and verified	OpenSSL (1.1.1f) rsyslogd 8.2001.0, acumen-tlsc-v2.2e, acumen-tlss-v2.2e, acumen-tlss, acumen-tls, X509-mod, acumen-sshs, Wireshark (V 3.6.7)	N/A
Console Server	Console Access	X.X.X.X	N/A	SSH	N/A	N/A	N/A	N/A
L3 Switch	L3 Switch	N/A	N/A	N/A	IOS	N/A	N/A	N/A
Gateway	Gateway	N/A	N/A	N/A	IOS	N/A	N/A	N/A
Bridge	Bridge	X.X.X.X	N/A	SSH	Linux pi-gmc 5.15.61-v8+	Manually set and verified	Wireshark Version 3.6.7	N/A

Virtual TOE:

Table 5 - Virtual Device Details

Device Details Table								
Device Details		**Network Details**			**System Details**			
Device Name	**Function**	**IP Address**	**MAC Address**	**Protocols**	**OS, including version**	**Timing Source**	**Software & Tools, including version**	**Credentials**
CM2500V	TOE	X.X.X.X	X:X:X:X:X:X	TLS/SSH	TRFEOS, 10.0.4	NTP Synced	N/A	N/A
User Laptop	Mgt. Access/Console Access	X.X.X.X	X:X:X:X:X:X	TLS/SSH	Windows 10	Manually set and verified	Chrome (Version 109.0.5414.120), Microsoft Edge (Version 110.0.1587.41), XCA (2.1.1) OpenSSL (1.1.1f) Putty (Release 0.77) Hex editor (Version 2.5.0.0)	N/A
ESXi Server/ Syslog Server/ OCSP Server/NTP Server	ubuntuVM / Syslog server/ OCSP server TLS client TLS server NTP server SSH Client	X.X.X.X	X:X:X:X:X:X	TLS/SSH	Ubuntu 20.04.4	Manually set and verified	OpenSSL (1.1.1f) rsyslogd 8.2001.0, acumen-tlsc-v2.2e, acumen-tlss-v2.2e, acumen-tlss, acumen-tls, X509-mod, acumen-sshs, Wireshark (V 3.6.7)	N/A

ESXi Server	ESXi server hosting TOE/ Console Access	X.X.X.X	N/A	TLS	7.0 and Intel(R) Xeon(R) CPU E5-4620 v4	N/A	N/A	N/A
L3 Switch	L3 Switch	N/A	N/A	N/A	IOS	N/A	N/A	N/A
Gateway	Gateway	N/A	N/A	N/A	IOS	N/A	N/A	N/A
Bridge	Bridge	X.X.X.X	N/A	SSH	Linux pi-gmc 5.15.61-v8+	Manually set and verified	Wireshark Version 3.6.7	N/A

4.3 TEST TIME AND LOCATION

All testing was carried out at Acumen Security office located at 2400 Research Blvd Suite #395, Rockville, MD 20850.Testing occurred from April 2023 to July 2024. Additionally, regression testing was performed in the months of September and October 2024.

The TOE was in a physically protected, access controlled, designated test lab with no unattended entry/exit ways. At the start of each day testing was performed, the test bed was verified to ensure that it was not compromised. All evaluation documentation was always kept in a secure repository.

5 DETAILED TEST CASES (TSS AND AGD ACTIVITIES)

5.1 MANDATORY REQUIREMENTS

5.1.1 SECURITY AUDIT (FAU)

5.1.1.1 FAU_GEN.1 AUDIT DATA GENERATION

5.1.1.1.1 FAU_GEN.1 TSS

For the administrative task of generating/import of, changing, or deleting of cryptographic keys as defined in FAU_GEN.1.1c, the TSS should identify what information is logged to identify the relevant key.

Evaluator Findings:
The evaluator examined the TSS row **FAU_GEN.1** and ensured that it identifies what information is logged to identify the relevant cryptographic key during generating/import, changing, or deleting. The relevant information is found in the following section(s): TOE Summary Specification **FAU_GEN.1**. Upon investigation, the evaluator found that the TSS states that: **For generating/importing of, changing, and deleting of certificates and associated keys, the TOE logs the certificate ID (SHA-1 Fingerprint) for TLS and "identity" for SSH which directly maps to a unique key pair.**

For distributed TOEs the evaluator shall examine the TSS and ensured that it describes which of the overall required auditable events defined in FAU_GEN.1.1 are generated and recorded by which TOE components.

Evaluator Findings:
The TOE is not a distributed TOE; hence this assurance activity is not applicable.

The evaluator shall ensure that the mapping of audit events to TOE components accounts for, and is consistent with, information provided in Table 1, as well as events in Tables 2, 4, and 5 (as applicable to the overall TOE). The evaluator confirmed that all components defined as generating audit information for a particular SFR contributed to that SFR as defined in the mapping of SFRs to TOE components, and that the audit records generated by each component covered all the SFRs that it implements.

Evaluator Findings:
The TOE is not a distributed TOE; hence this assurance activity is not applicable.

Verdict:
PASS.

5.1.1.1.2 FAU_GEN.1 AGD

The evaluator shall check the guidance documentation and ensure that it provides an example of each auditable event required by FAU_GEN.1 (i.e. at least one instance of each auditable event, comprising the mandatory, optional and selection-based SFR sections as applicable, shall be provided from the actual audit record).

Evaluator Findings:
The evaluator checked the AGD and ensured that it provides an example of each auditable event required by FAU_GEN.1 (i.e. at least one instance of each auditable event, comprising the mandatory, optional and selection-based SFR sections as applicable, was provided from the actual audit record). The relevant information is found in the following section(s): **CC-NDcPP Events** Upon investigation, the evaluator found that the AGD provides an example of each auditable event required by FAU_GEN.1

The evaluator shall also make a determination of the administrative actions related to TSF data related to configuration changes.

Evaluator Findings:

The evaluator made a determination of the administrative actions related to TSF data related to configuration changes.

The relevant information is found in the following section(s): **CC-NDcPP Events** gives information about log format and the below sections give information about configuration steps.

TSF activity	**AGD Section**
Time change	Setting Time
Addition of certificate	Addition of Certificates to Trust Store
Removal of certificate	Removal of Certificates from Trust Store
Generating/import of, changing, or deleting of cryptographic keys	Configuring X.509 certificate Authentication for the Web UI
Resetting passwords	Resetting Passwords

Upon investigation, the evaluator found that the administrative actions related to TSF data related to configuration changes are mentioned in the AGD.

The evaluator shall examine the guidance documentation and make a determination of which administrative commands, including subcommands, scripts, and configuration files, are related to the configuration (including enabling or disabling) of the mechanisms implemented in the TOE that are necessary to enforce the requirements specified in the cPP.

Evaluator Findings:
The evaluator examined the AGD and made a determination of which administrative commands, including subcommands, scripts, and configuration files, are related to the configuration (including

enabling or disabling) of the mechanisms implemented in the TOE that are necessary to enforce the requirements specified in the cPP.

The relevant information is found in the following section(s): Entire AGD

Upon investigation, the evaluator found that the AGD states that the following are applicable:

Administrative Activity	**Method (Command/GUI Configuration)**	**Section**
User Creation	Command Line Interface Graphical User Interface	Sections titled: **'User Creation via the Web UI.'** **'User Creation via the CLI'**
Audit configuration	Command Line Interface	Section titled: '**Audit Server Configuration**'
Authentication failure configuration	Command Line Interface	Section titled: '**Authentication failure Handling'**
Setting time	Command Line Interface	Section titled: **'Setting Time'**
Software update	Command Line Interface	Section titled: **'Software Updates'**
Configuring banner	Command Line Interface Graphical User Interface	Section titled: **'Customizing Login Banners and Messages Using the Web UI'** **'Customizing Login Banners and Messages Using the CLI'**

The evaluator shall document the methodology or approach taken while determining which actions in the administrative guide are related to TSF data related to configuration changes. The evaluator may perform this activity as part of the activities associated with ensuring that the corresponding guidance documentation satisfies the requirements related to it.

Evaluator Findings:

The evaluator documented the methodology or approach taken while determining which actions in the administrative guide are related to TSF data related to configuration changes. The evaluator may perform this activity as part of the activities associated with ensuring that the corresponding AGD satisfies the requirements related to it.

The relevant information is found in the following section(s): Entire AGD

Upon investigation, the evaluator found that the AGD states that:

Administrative Activity	Method (Command/GUI Configuration)	Test Case(s)
Audit configuration	CLI	FAU_STG_EXT.1_Test 2
User Creation	GUI/CLI	FIA_PMG_EXT.1_Test 1
Authentication failure configuration	CLI	FIA_AFL_EXT.1_Test 1
Software update	CLI	FPT_TUD_EXT.1 Test #2 (b)
Setting time	CLI	FPT_STM.1.1_Test 1
Configuring banner	GUI/CLI	FTA_TAB.1_Test 1

Verdict:
PASS.

5.1.1.2 FAU_GEN.2 USER IDENTITY ASSOCIATION

5.1.1.2.1 TSS & AGD

The TSS and Guidance Documentation requirements for FAU_GEN.2 are already covered by the TSS and AGD requirements for FAU_GEN.1.

5.1.1.3 FAU_STG_EXT.1 PROTECTED AUDIT EVENT STORAGE

5.1.1.3.1 FAU_STG_EXT.1 TSS

The evaluator shall examine the TSS to ensure it describes the means by which the audit data are transferred to the external audit server, and how the trusted channel is provided.

Evaluator Findings:

The evaluator examined the TSS row **FAU_STG_EXT.1** and ensured that it describes the means by which the audit data are transferred to the external audit server, and how the trusted channel is provided.

The relevant information is found in the following section(s): TOE Summary Specification **FAU_STG_EXT.1**.

Upon investigation, the evaluator found that the TSS states that: **The TOE can be configured to export syslog records to a specified, external syslog server. The TOE protects communications with an**

external syslog server via TLS. The TOE transmits its audit events to all configured syslog servers in real-time.

The evaluator shall examine the TSS to ensure it describes the amount of audit data that are stored locally; what happens when the local audit data store is full; and how these records are protected against unauthorized access.

Evaluator Findings:
The evaluator examined the TSS and ensured it describes the amount of audit data that are stored locally; what happens when the local audit data store is full; and how these records are protected against unauthorized access. The relevant information is found in the following section(s): TOE Summary Specification **FAU_STG_EXT.1**. Upon investigation, the evaluator found that the TSS states that: **The amount of audit data that can be stored locally is configurable by setting the local log rotation parameters (e.g. see the logging files rotation CLI commands). The TOE defaults to rotating the log file when it reaches 256MB and retaining 40 compressed archives. This results in storing 10.25GB of uncompressed logs.** **When the local log is full, the oldest archive file is deleted to allow a new log to be created.** **Local audit records are stored in a directory that does not allow administrators to modify the contents.**

The evaluator shall examine the TSS to ensure it describes whether the TOE is a standalone TOE that stores audit data locally or a distributed TOE that stores audit data locally on each TOE component or a distributed TOE that contains TOE components that cannot store audit data locally on themselves but need to transfer audit data to other TOE components that can store audit data locally.

Evaluator Findings:
The TOE is not a distributed TOE. The TSS row **FAU_STG_EXT.1** states that the **TOE is a standalone TOE that stores audit data locally.**

The evaluator shall examine the TSS to ensure that for distributed TOEs it contains a list of TOE components that store audit data locally.

Evaluator Findings:
The TOE is not a distributed TOE hence this assurance activity is not applicable.

The evaluator shall examine the TSS to ensure that for distributed TOEs that contain components which do not store audit data locally but transmit their generated audit data to other components it contains a mapping between the transmitting and storing TOE components.

Evaluator Findings:
The TOE is not a distributed TOE; hence this assurance activity is not applicable.

The evaluator shall examine the TSS to ensure that it details the behaviour of the TOE when the storage space for audit data is full. When the option 'overwrite previous audit record' is selected this description should include an outline of the rule for overwriting audit data. If 'other actions' are chosen such as sending the new audit data to an external IT entity, then the related behaviour of the TOE shall also be detailed in the TSS.

Evaluator Findings:
The evaluator examined the TSS row **FAU_STG_EXT.1** and ensured that it details the behaviour of the TOE when the storage space for audit data is full. When the option 'overwrite previous audit record' is selected this description should include an outline of the rule for overwriting audit data. If 'other actions' are chosen such as sending the new audit data to an external IT entity, then the related behaviour of the TOE is detailed in the TSS. The relevant information is found in the following section(s): TOE Summary Specification **FAU_STG_EXT.1**. Upon investigation, the evaluator found that: **The TOE defaults to rotating the log file when it reaches 256MB and retaining 40 compressed archives. This results in storing 10.25GB of uncompressed logs. When the local log is full, the oldest archive file is deleted to allow a new log to be created.** **When the local storage space for audit data is full, the oldest archive file is deleted to allow a new log to be created so the TOE overwrites previous audit records.** **The 'other actions' is not claimed in the ST.**

The evaluator shall examine the TSS to ensure that it details whether the transmission of audit information to an external IT entity can be done in real- time or periodically. In case the TOE does not perform transmission in real- time the evaluator needs to verify that the TSS provides details about what event stimulates the transmission to be made as well as the possible acceptable frequency for the transfer of audit data.

Evaluator Findings:
The evaluator examined the TSS row **FAU_STG_EXT.1** and ensured that it details whether the transmission of audit information to an external IT entity can be done in real-time, periodically, or both. In the case where the TOE is capable of performing transmission periodically, the evaluator verified that the TSS provides details about what event stimulates the transmission to be made as well as the possible acceptable frequency for the transfer of audit data. The relevant information is found in the following section(s): TOE Summary Specification **FAU_STG_EXT.1**.

Upon investigation, the evaluator found that the TSS states that: **The TOE transmits its audit events to all configured syslog servers in real-time.**

For distributed TOEs the evaluator shall examine the TSS to ensure it describes to which TOE components this SFR applies and how audit data transfer to the external audit server is implemented among the different TOE components (e.g. every TOE components does its own transfer or the data is sent to another TOE component for central transfer of all audit events to the external audit server).

Evaluator Findings:
The TOE is not a distributed; TOE hence this assurance activity is not applicable.

For distributed TOEs the evaluator shall examine the TSS to ensure it describes which TOE components are storing audit information locally and which components are buffering audit information and forwarding the information to another TOE component for local storage. For every component the TSS shall describe the behaviour when local storage space or buffer space is exhausted.

Evaluator Findings:
The TOE is not a distributed; TOE hence this assurance activity is not applicable.

Verdict:
PASS.

5.1.1.3.2 FAU_STG_EXT.1 AGD

The evaluator shall also examine the guidance documentation to ensure it describes how to establish the trusted channel to the audit server, as well as describe any requirements on the audit server (particular audit server protocol, version of the protocol required, etc.), as well as configuration of the TOE needed to communicate with the audit server.

Evaluator Findings:
The evaluator examined the guidance documentation **section 5** and ensured it describes how to establish the trusted channel to the audit server, as well as describe any requirements on the audit server (particular audit server protocol, version of the protocol required, etc.), as well as configuration of the TOE needed to communicate with the audit server. The relevant information is found in the following section(s): **Audit Server Requirements** and **Audit Server Configuration** Upon investigation, the evaluator found that the AGD states that: **TOE establishes the trusted channel to the audit server using below command:** **'logging <Ip address> protocol tls port <6514>'** **The device will begin sending audit events to the audit server as soon as the connection is made after the audit server is configured. If the server certificate is invalid, the TSF will by default not create a trusted channel.** **AGD describes the TOE configuration needed to communicate with the audit server and states that the audit server must be a Syslog server that supports TCP and TLS 1.2.**

The evaluator shall also examine the guidance documentation to determine that it describes the relationship between the local audit data and the audit data that are sent to the audit log server. For example, when an audit event is generated, is it simultaneously sent to the external server and the local store, or is the local store used as a buffer and "cleared" periodically by sending the data to the audit server.

Evaluator Findings:
The evaluator also examined the guidance documentation **section 5.2** and determined that it describes the relationship between the local audit data and the audit data that are sent to the audit log server. The relevant information is found in the following section(s): **System Behavior** Upon investigation, the evaluator found that the AGD states that: **When configured to use an audit server the FireEye AX, CM, EX, FX, HX, NX, and VX Series Appliances transmit audit events to the audit server in real-time.**

The evaluator shall also ensure that the guidance documentation describes all possible configuration options for FAU_STG_EXT.1.3 and the resulting behaviour of the TOE for each possible configuration. The description of possible configuration options and resulting behaviour shall correspond to those described in the TSS.

Evaluator Findings:
The evaluator ensured that the AGD describes all possible configuration options for FAU_STG_EXT.1.3 and the resulting behaviour of the TOE for each possible configuration. The description of possible configuration options and resulting behaviour correspond to those described in the TSS. The relevant information is found in the following section(s): **System Behavior** Upon investigation, the evaluator found that the AGD states that: **The amount of audit data that can be stored locally is configurable by setting the local log rotation parameters using the following command.** **'logging files rotation criteria size <log file size threshold>'** **When the local log is full, the oldest archive file is deleted to allow a new log to be created so the TOE overwrites previous audit records.** Next, the evaluator compared the exhausted local audit handling description found in AGD to the description provided by the TSS of the ST. The descriptions of the behavior found in AGD and ST are consistent.

Verdict:
PASS.

5.1.2 CRYPTOGRAPHIC SUPPORT (FCS)

5.1.2.1 FCS_CKM.1 CRYPTOGRAPHIC KEY GENERATION

5.1.2.1.1 FCS_CKM.1 TSS

The evaluator shall ensure that the TSS identifies the key sizes supported by the TOE.

Evaluator Findings:
The evaluator ensured that the TSS row **FCS_CKM.1** identifies the key sizes supported by the TOE. The relevant information is found in the following section(s): TOE Summary Specification **FCS_CKM.1**. Upon investigation, the evaluator found that the TSS states that: **The TOE supports RSA key generation schemes as specified in FIPS 186-4, with key sizes of 2048 and 3072 bits.** **The TOE supports Elliptic Curve key generation of P-256, P-384, P-521.** **The TOE supports DHG14(2048 bits) key generation in support of DH key exchanges as part of TLS.** **The TOE supports DHG14(2048 bits), DH16(4096) and DH18(8192) key generation in support of DH key exchanges as part of SSH.**

If the ST specifies more than one scheme, the evaluator shall examine the TSS to verify that it identifies the usage for each scheme.

Evaluator Findings:
The evaluator examined the TSS **FCS_CKM.1** and verified that it identifies the usage for each scheme. The relevant information is found in the following section(s): TOE Summary Specification **FCS_CKM.1**. Upon investigation, the evaluator found that the TSS states that: **The RSA keys are used in support of digital certificates and keyed authentication for TLS and SSH.** **The Elliptic Curve keys are used in support of ECDH key exchange as part of TLS.** **The TOE supports DHG14(2048 bits) key generation in support of DH key exchanges as part of TLS.** **The TOE supports DHG14(2048 bits), DH16(4096) and DH18(8192) key generation in support of DH key exchanges as part of SSH.**

Verdict:
PASS.

5.1.2.1.2 FCS_CKM.1 AGD

The evaluator shall verify that the AGD guidance instructs the administrator how to configure the TOE to use the selected key generation scheme(s) and key size(s) for all cryptographic protocols defined in the Security Target.

Evaluator Findings:

The evaluator verified that the AGD instruct the administrator how to configure the TOE to use the selected key generation scheme(s) and key size(s) for all cryptographic protocols defined in the Security Target.
 	
The relevant information is found in the following section(s): **Details of CC Mode**
 	
Upon investigation, the evaluator found that the AGD states that:
 	
Once NDcPP compliance is enabled below settings will be applied by default without any additional configuration changes.
 	
• **Appliance supports signature generation and verification for RSA (2048 and 3072 bits) and ECDSA (P-256, P-384, P-521), in accordance with FIPS PUB 186-4**
  ○ **RSA signature generation and verification are used for the TLS and SSH protocols.**
  ○ **ECDSA signature verification is used in TLS.**
• **Appliance provides DHG14(2048 bits) key generation in support of DH key exchanges as part of TLS.**

Appliance provides key generation for DHG14 (2048 bits), DH16 (4096 bits), and DH18 (8192 bits) in DH key exchanges used in SSH.

Verdict:
PASS.

5.1.2.2 FCS_CKM.2 CRYPTOGRAPHIC KEY ESTABLISHMENT

5.1.2.2.1 FCS_CKM.2 TSS **[TD0580]**

The evaluator shall ensure that the supported key establishment schemes correspond to the key generation schemes identified in FCS_CKM.1.1.

Evaluator Findings:
The evaluator ensured that the supported key establishment schemes correspond to the key generation schemes identified in FCS_CKM.1.1. The relevant information is found in the following section(s): TOE Summary Specification **FCS_CKM.2.** Upon investigation, the evaluator found that : **In support of secure cryptographic protocols, the TOE supports several key establishment schemes, including:** • **ECC based key exchange based on NIST SP 800-56Ar3;** • **FFC based key exchange based on NIST SP 800-56Ar3;** • **FFC using 'safe-prime' based key exchange based on NIST SP 800-56Ar3** **The evaluator determined that the supported key establishment schemes correspond to the key generation schemes identified in CKM.1.**

If the ST specifies more than one scheme, the evaluator shall examine the TSS to verify that it identifies the usage for each scheme. It is sufficient to provide the scheme, SFR, and service in the TSS.

The intent of this activity is to be able to identify the scheme being used by each service. This would mean, for example, one way to document scheme usage could be:.

Scheme	SFR	Service
RSA	FCS_TLSS_EXT.1	Administration
ECDH	FCS_SSHC_EXT.1	Audit Server
ECDH	FCS_IPSEC_EXT.1	Authentication Server

The information provided in the example above does not necessarily have to be included as a table but can be presented in other ways as long as the necessary data is available.

Evaluator Findings:

The evaluator examined the TSS row **FCS_CKM.2** to verify that it identifies the usage for each scheme. It is sufficient to provide the scheme, SFR, and service in the TSS.

The relevant information is found in the following section(s): TOE Summary Specification **FCS_CKM.2**.

Upon investigation, the evaluator found that the that: **The TOE supports several key establishment schemes, including:**

- **ECC based key exchange based on NIST SP 800-56Ar3;**
- **FFC based key exchange based on NIST SP 800-56Ar3;**
- **FFC using 'safe-prime' based key exchange based on NIST SP 800-56Ar3**

Scheme	SFRs	Service
ECC	FCS_TLSC_EXT.1	Syslog
	FCS_TLSS_EXT.1	Remote Administration
FFC	FCS_TLSC_EXT.1	Syslog
	FCS_TLSS_EXT.1	Remote Administration
FFC Safe Primes	FCS_TLSC_EXT.1	Audit Server
	FCS_TLSS_EXT.1 FCS_SSHS_EXT.1	Remote Administration

Note: VX series models don't support Web UI Feature and hence SFR FCS_TLSS_EXT.1 is not applicable.

FFC safe Primes (DH Group 14, DH Group 16 and DH Group 18) are used in SSH. DH Groups 14, 16 and 18 are used for implementing SSH which protects the remote management session between the remote management workstation and the TOE.

Verdict:
PASS.

5.1.2.2.2 FCS_CKM.2 AGD

The evaluator shall verify that the AGD guidance instructs the administrator how to configure the TOE to use the selected key establishment scheme(s).

Evaluator Findings:
The evaluator verified that the AGD guidance instructs the administrator how to configure the TOE to use the selected key establishment scheme(s). The relevant information is found in the following section(s): **Details of CC Mode** Upon investigation, the evaluator found the information in AGD and summarized it as: **Once NDcPP compliance is enabled below settings will be applied by default without any additional configuration changes.** • **Appliance supports signature generation and verification for ECDSA (P-256, P-384, P-521), in accordance with FIPS PUB 186-4.**   ○ **ECDSA signature verification is used in TLS** • **Appliance provides DHG14(2048 bits) key generation in support of DH key exchanges as part of TLS.** • **Appliance provides key generation for DHG14 (2048 bits), DH16 (4096 bits), and DH18 (8192 bits) in DH key exchanges used in SSH.**

Verdict:
PASS.

5.1.2.3 FCS_CKM.4 CRYPTOGRAPHIC KEY DESTRUCTION

5.1.2.3.1 FCS_CKM.4 TSS

The evaluator examines the TSS to ensure it lists all relevant keys (describing the origin and storage location of each), all relevant key destruction situations (e.g. factory reset or device wipe function, disconnection of trusted channels, key change as part of a secure channel protocol), and the destruction method used in each case. For the purpose of this Evaluation Activity the relevant keys are those keys that are relied upon to support any of the SFRs in the Security Target. The evaluator confirms that the description of keys and storage locations is consistent with the functions carried out by the TOE (e.g. that all keys for the TOE-specific secure channels and protocols, or that support FPT_APW.EXT.1 and FPT_SKP_EXT.1, are accounted for). In particular, if a TOE claims not to store plaintext keys in non-volatile memory then the evaluator checks that this is consistent with the operation of the TOE.

Evaluator Findings:

The evaluator examined the TSS row **FCS_CKM.4** and section **Cryptographic Key Destruction** to ensure it lists all relevant keys (describing the origin and storage location of each), all relevant key destruction situations (e.g. factory reset or device wipe function, disconnection of trusted channels, key change as part of a secure channel protocol), and the destruction method used in each case.

The relevant information is found in the following section(s): TOE Summary Specification **FCS_CKM.4**.

Upon investigation, the evaluator found that: '**Table16-Key Zeroization**' from the ST section **Cryptographic Key Destruction** contains a column dedicated to the origin of the key, type of the key, the storage of the key, and method of zeroization.

Non-volatile keys are overwritten with zeros using a single pass when the administrator disables CC mode. As part of the disablement function, the device is power cycled to zeroize keys in volatile memory.

TSS states that: All keys from non-volatile memory are stored plaintext and are ACL protected from unauthorized access as described in FPT_SKP_EXT.1 and the Storage/Protection column. The TSF meets all requirements specified in the NDcPPv2.2e for destruction of keys.

The evaluator shall check to ensure the TSS identifies how the TOE destroys keys stored as plaintext in non-volatile memory, and that the description includes identification and description of the interfaces that the TOE uses to destroy keys (e.g., file system APIs, key store APIs).

Evaluator Findings:

The evaluator confirmed that the description of keys and storage locations is consistent with the functions carried out by the TOE (e.g. that all keys for the TOE-specific secure channels and protocols, or that support FPT_APW.EXT.1 and FPT_SKP_EXT.1, are accounted for). In particular, a TOE claims to store plaintext keys in non-volatile memory.

The relevant information is found in the following section(s): TOE Summary Specification **FCS_CKM.4 and Cryptographic Key Destruction**.

Upon investigation, the evaluator found that the TSS states that: The TSF meets all requirements specified in the NDcPPv2.2e for destruction of keys. All keys within the TSF are securely destroyed as per the descriptions given in Table 16 in the ST.

The evaluator examined the section titled section **Cryptographic Key Destruction** in the Security Target and found that the column titled **'Method of Zeroization'** (overwriting by zeros) and **'Keys/CSP Storage location'** gives detail information about how the TOE destroys keys stored as plaintext in non-volatile memory and the interface used to zeroize (either the compliance zeroize command or as part of a session closing) is described. RAM is volatile storage location and ACL protected directories are non-volatile storage location.

Note that where selections involve 'destruction of reference' (for volatile memory) or 'invocation of an interface' (for non-volatile memory) then the relevant interface definition is examined by the evaluator to ensure that the interface supports the selection(s) and description in the TSS. In the case of non-volatile memory, the evaluator includes in their examination the relevant interface description for each media type on which plaintext keys are stored. The presence of OS-level and storage device-level swap and cache files is not examined in the current version of the Evaluation Activity.

Evaluator Findings:
The evaluator checked to ensure the TSS row **FCS_CKM.4** and section **Cryptographic Key Destruction** in the ST identifies how the TOE destroys keys stored as plaintext in non-volatile memory, and that the description includes identification and description of the interfaces that the TOE uses to destroy keys (e.g., file system APIs, key store APIs). The relevant information is found in the following section(s): TOE Summary Specification **FCS_CKM.4** and section **Cryptographic Key Destruction.** Upon investigation, the evaluator found that: **The TSF meets all requirements specified in the NDcPPv2.2e for destruction of keys. All keys within the TSF are securely destroyed as per the descriptions given in Table 16 in the ST.** The Keys can be zeroize using the "compliance declassify zeroize" command. ST does not select 'destruction of reference'.

Where the TSS identifies keys that are stored in a non-plaintext form, the evaluator shall check that the TSS identifies the encryption method and the key-encrypting-key used, and that the key-encrypting-key is either itself stored in an encrypted form or that it is destroyed by a method included under FCS_CKM.4.

Evaluator Findings:
The TSS row **FCS_CKM.4** identifies no keys that are stored in a non-plaintext form. The relevant information is found in the following section(s): TOE Summary Specification **FCS_CKM.4.** Upon investigation, the evaluator found that the TSS states that: **All keys are stored plaintext and are protected from unauthorized access as described in FPT_SKP_EXT.1.**

The evaluator shall check that the TSS identifies any configurations or circumstances that may not conform to the key destruction requirement (see further discussion in the Guidance Documentation section below). Note that reference may be made to the Guidance Documentation for description of the detail of such cases where destruction may be prevented or delayed.

Evaluator Findings:
The evaluator checked that the TSS row **FCS_CKM.4** identifies any configurations or circumstances that may not conform to the key destruction requirement (see further discussion in the Guidance Documentation section below).

The relevant information is found in the following section(s): TOE Summary Specification **FCS_CKM.4** Upon investigation, the evaluator found that the TOE does not have any circumstances that may not conform to key destruction requirements.

Where the ST specifies the use of "a value that does not contain any CSP" to overwrite keys, the evaluator examines the TSS to ensure that it describes how that pattern is obtained and used, and that this justifies the claim that the pattern does not contain any CSPs.

Evaluator Findings:
The ST does not specify the use of "a value that does not contain any CSP" to overwrite keys.

Verdict:
PASS.

5.1.2.3.2 FCS_CKM.4 AGD

A TOE may be subject to situations that could prevent or delay key destruction in some cases. The evaluator shall check that the guidance documentation identifies configurations or circumstances that may not strictly conform to the key destruction requirement, and that this description is consistent with the relevant parts of the TSS (and any other supporting information used).

Evaluator Findings:
The evaluator checked that the guidance documentation **section 8** identifies configurations or circumstances that may not strictly conform to the key destruction requirement, and that this description is consistent with the relevant parts of the TSS (and any other supporting information used). The relevant information is found in the following section(s): **Zeroization** Upon investigation, the evaluator reviewed **the AGD documentation for the TOE and found no items that did not meet conformance to the key destruction requirement, and found that it is consistent with the relevant part of the TSS.**

The evaluator shall check that the guidance documentation provides guidance on situations where key destruction may be delayed at the physical layer.

For example, when the TOE does not have full access to the physical memory, it is possible that the storage may be implementing wear-levelling and garbage collection. This may result in additional copies of the key that are logically inaccessible but persist physically. Where available, the TOE might then describe use of the TRIM command3 and garbage collection to destroy these persistent copies upon their deletion (this would be explained in TSS and Operational Guidance).

Evaluator Findings:
The evaluator checked that the guidance documentation section 8 provides guidance on situations where key destruction may be delayed at the physical layer.

The relevant information is found in the following section(s): **Zeroization**

Upon investigation, the evaluator found that the AGD states that: **There is no situation that could prevent or delay key destruction.**

Verdict:
PASS.

5.1.2.4 FCS_COP.1/DATAENCRYPTION CRYPTOGRAPHIC OPERATION (AES DATA ENCRYPTION/DECRYPTION)

5.1.2.4.1 FCS_COP.1/DATAENCRYPTION TSS

The evaluator shall examine the TSS to ensure it identifies the key size(s) and mode(s) supported by the TOE for data encryption/decryption.

Evaluator Findings:
The evaluator examined the TSS row **FCS_COP.1/DataEncryption** to ensure it identifies the key size(s) and mode(s) supported by the TOE for data encryption/decryption. The relevant information is found in the following section(s): TOE Summary Specification **FCS_COP.1/DataEncryption**. Upon investigation, the evaluator found that the TSS states that: **The TOE provides symmetric encryption and decryption capabilities using 128-bit and 256-bit AES as specified in ISO 18033-3, in CBC mode and CTR mode as described in ISO 10116 and GCM mode as described in ISO 19772. AES is implemented in the following protocols: TLS and SSH.**

Verdict:
PASS.

5.1.2.4.2 FCS_COP.1/DATAENCRYPTION AGD

The evaluator shall verify that the AGD guidance instructs the administrator how to configure the TOE to use the selected mode(s) and key size(s) defined in the Security Target supported by the TOE for data encryption/decryption.

Evaluator Findings:
The evaluator verified that the AGD guidance instructs the administrator how to configure the TOE to use the selected mode(s) and key size(s) defined in the Security Target supported by the TOE for data encryption/decryption. The relevant information is found in the following section(s): **Details of CC Mode**

Upon investigation, the evaluator found that the AGD states that:
Once NDcPP compliance is enabled below settings will be applied by default without any additional configuration changes.

- **Appliance provides AES encryption/decryption in CBC, CTR an GCM mode with 128-bit and 256-bit keys.**
  - **AES is implemented in the following protocols: TLS and SSH**

Verdict:
PASS.

5.1.2.5 FCS_COP.1/SIGGEN CRYPTOGRAPHIC OPERATION (SIGNATURE GENERATION AND VERIFICATION)

5.1.2.5.1 FCS_COP.1/SIGGEN TSS

The evaluator shall examine the TSS to determine that it specifies the cryptographic algorithm and key size supported by the TOE for signature services.

Evaluator Findings:
The evaluator examined the TSS row **FCS_COP.1/SigGen** to determine that it specifies the cryptographic algorithm and key size supported by the TOE for signature services. The relevant information is found in the following section(s): TOE Summary Specification **FCS_COP.1/SigGen**. Upon investigation, the evaluator found that the TSS states that: **The TOE provides cryptographic signature generation and verification services using:** • **RSA Signature Algorithm with key size of 2048 bits or 3072 bits,** • **ECDSA Signature Algorithm with NIST curves P-256, P-384 and P-521.**

Verdict:
PASS.

5.1.2.5.2 FCS_COP.1/SIGGEN AGD

The evaluator shall verify that the AGD guidance instructs the administrator how to configure the TOE to use the selected cryptographic algorithm and key size defined in the Security Target supported by the TOE for signature services.

Evaluator Findings:
The evaluator verified that the AGD guidance instructs the administrator how to configure the TOE to use the selected cryptographic algorithm and key size defined in the Security Target supported by the TOE for signature services. The relevant information is found in the following section(s): **Details of CC Mode**

Upon investigation, the evaluator found that the AGD states that:

Once NDcPP compliance is enabled below settings will be applied by default without any additional configuration changes.

- **Appliance supports signature generation and verification for RSA (2048 and 3072 bits) and ECDSA (P-256, P-384, P-521), in accordance with FIPS PUB 186-4.**
  - **RSA signature generation and verification are used for the TLS and SSH protocols**
  - **ECDSA signature verification is used in TLS**
- **Appliance provides DHG14(2048 bits) key generation in support of DH key exchanges as part of TLS.**
- **Appliance provides key generation for DHG14 (2048 bits), DH16 (4096 bits), and DH18 (8192 bits) in DH key exchanges used in SSH.**

Verdict:
PASS.

5.1.2.6 FCS_COP.1/HASH CRYPTOGRAPHIC OPERATION (HASH ALGORITHM)

5.1.2.6.1 FCS_COP.1/HASH TSS

The evaluator shall check that the association of the hash function with other TSF cryptographic functions (for example, the digital signature verification function) is documented in the TSS.

Evaluator Findings:

The evaluator checked that the association of the hash function with other TSF cryptographic functions (for example, the digital signature verification function) is documented in the TSS row **FCS_COP.1/Hash**.

The relevant information is found in the following section(s): TOE Summary Specification **FCS_COP.1/Hash**.

Upon investigation, the evaluator found that the TSS states that: **The TOE provides cryptographic hashing services using SHA-1, SHA-256, SHA-384, and SHA-512 as specified in ISO/IEC 10118-3:2004 which are implemented in the following parts of the TSF:**

- **NTP – SHA1**
- **TLS and SSH - SHA1, SHA-256, SHA-384, SHA-512;**
- **Digital signature verification as part of trusted update validation - SHA-256**
- **Hashing of passwords in non-volatile storage - SHA-512**
- **Conditioning entropy data – SHA-512**

Verdict:
PASS.

5.1.2.6.2 FCS_COP.1/HASH AGD

The evaluator checks the AGD documents to determine that any configuration that is required to configure the required hash sizes is present.

Evaluator Findings:
The evaluator checked the AGD documents to determine that any configuration that is required to configure the required hash sizes is present. The relevant information is found in the following section(s): **Details of CC Mode** Upon investigation, the evaluator found that the AGD states that: **Once NDcPP compliance is enabled below settings will be applied by default without any additional configuration changes.** • **It provides cryptographic hashing services for key generation using SHA-1, SHA-256, SHA-384, and SHA-512 as specified in ISO/IEC 10118-3:2004.** ○ **NTP – SHA1** ○ **TLS and SSH - SHA1, SHA-256, SHA-384 and SHA-512** ○ **Digital signature verification as part of trusted update validation - SHA-256** ○ **Hashing of passwords in non-volatile storage - SHA-512** ○ **Conditioning entropy data – SHA-512**

Verdict:
PASS.

5.1.2.7 FCS_COP.1/KEYEDHASH CRYPTOGRAPHIC OPERATION (KEYED HASH ALGORITHM)

5.1.2.7.1 FCS_COP.1/KEYEDHASH TSS

The evaluator shall examine the TSS to ensure that it specifies the following values used by the HMAC function: key length, hash function used, block size, and output MAC length used.

Evaluator Findings:
The evaluator examined the TSS **FCS_COP.1/KeyedHash** to ensure that it specifies the following values used by the HMAC function: key length, hash function used, block size, and output MAC length used. The relevant information is found in the following section(s): TOE Summary Specification **FCS_COP.1/KeyedHash.**

Upon investigation, the evaluator found that the TSS states that: **The characteristics of the HMACs used in the TOE are given in the following table:**

Algorithm	Hash function	Block size	Key size	Digest size
HMAC-SHA-1	SHA-1	512 bits	160 bits	160 bits
HMAC-SHA-256	SHA-256	512 bits	256 bits	256 bits
HMAC-SHA-384	SHA-384	1024 bits	384 bits	384 bits
HMAC-SHA-512	SHA-512	1024 bits	512 bits	512 bits

Verdict:
PASS.

5.1.2.7.2 FCS_COP.1/KEYEDHASH AGD

The evaluator shall verify that the AGD guidance instructs the administrator how to configure the TOE to use the values used by the HMAC function: key length, hash function used, block size, and output MAC length used defined in the Security Target supported by the TOE for keyed hash function.

Evaluator Findings:

The evaluator verified that the AGD guidance instructs the administrator how to configure the TOE to use the values used by the HMAC function: key length, hash function used, block size, and output MAC length used defined in the Security Target supported by the TOE for keyed hash function.

The relevant information is found in the following section(s): **Details of CC Mode**

Upon investigation, the evaluator found that the AGD states that:
Once NDcPP compliance is enabled below settings will be applied by default without any additional configuration changes.

- **Appliance implements HMAC message authentication. HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 are supported with cryptographic key sizes of 160, 256, 384, and 512 bits and message digest sizes of 160, 256, 384, and 512 bits.**
  - **HMAC is implemented in the following protocols: TLS and SSH**

Verdict:
PASS.

5.1.2.8 FCS_RBG_EXT.1 EXTENDED: CRYPTOGRAPHIC OPERATION (RANDOM BIT GENERATION)

5.1.2.8.1 FCS_RBG_EXT.1 TSS

The evaluator shall examine the TSS to determine that it specifies the DRBG type, identifies the entropy source(s) seeding the DRBG, and state the assumed or calculated min-entropy supplied either separately by each source or the min-entropy contained in the combined seed value.

Evaluator Findings:
The evaluator examined the TSS row **FCS_RBG_EXT.1** and determined that it specifies the DRBG type, identifies the entropy source(s) seeding the DRBG, and state the assumed or calculated min-entropy supplied either separately by each source or the min- entropy contained in the combined seed value. The relevant information is found in the following section(s): TOE Summary Specification **FCS_RBG_EXT.1**. Upon investigation, the evaluator found that the TSS states that**: The TOE implements a NIST-approved CTR_DRBG(AES-256) and HMAC_DRBG(SHA-512), as specified in SP 800-90A.** **The entropy source used to seed the Deterministic Random Bit Generator is a random set of bits supplied from one software noise source. (This ST considers the sources 'software' simply because the entropy sources are not considered True Random Number Generators (TRNGs) based on random properties of physical processes.) The 512-bit seed value contains at least 256 bits of entropy.**

Verdict:
PASS.

5.1.2.8.2 FCS_RBG_EXT.1 AGD

The evaluator shall confirm that the guidance documentation contains appropriate instructions for configuring the RNG functionality.

Evaluator Findings:
The evaluator confirmed that the guidance documentation contains appropriate instructions for configuring the RNG functionality. The relevant information is found in the following section(s): **Details of CC Mode** Upon investigation, the evaluator found that the AGD states that: **Once NDcPP compliance is enabled below settings will be applied by default without any additional configuration changes.** • **Appliance provides NIST-approved CTR_DRBG(AES-256) and HMAC_DRBG(SHA-512), as specified in SP 800-90A for RNG functionality.**

Verdict:
PASS.

5.1.3 IDENTIFICATION AND AUTHENTICATION (FIA)

5.1.3.1 FIA_AFL.1 AUTHENTICATION FAILURE MANAGEMENT

5.1.3.1.1 FIA_AFL.1 TSS

The evaluator shall examine the TSS to determine that it contains a description, for each supported method for remote administrative actions, of how successive unsuccessful authentication attempts are detected and tracked. The TSS shall also describe the method by which the remote administrator is prevented from successfully logging on to the TOE, and the actions necessary to restore this ability.

Evaluator Findings:
The evaluator shall examine the TSS row **FIA_AFL.1** to determine that it contains a description, for each supported method for remote administrative actions, of how successive unsuccessful authentication attempts are detected and tracked. The TSS shall also describe the method by which the remote administrator is prevented from successfully logging on to the TOE, and the actions necessary to restore this ability. The relevant information is found in the following section(s): TOE Summary Specification, **FIA_AFL.1**. Upon investigation, the evaluator found that the TSS states that: **The TOE is capable of tracking authentication failures for each of the claimed authentication mechanisms (username/password, SSH public key) for SSH administration method and claimed authentication mechanisms (username/password) for GUI[4] administration method.** **The administrator can configure the maximum number of failed attempts using the CLI interface via the aaa authentication attempts command. The configurable range is between 1 and 15 attempts.** **When a user account has sequentially failed authentication the configured number of times, the account will be locked. The locking mechanism can be configured to remain locked until an administrator unlocks the account, or it can be configured to unlock after a specified period of time.** **If the administrator is required to intervene to unlock an account, this is done using the CLI via the aaa authentication attempts reset CLI command. The aaa authentication attempts commands apply to authentication attempts through both SSH and the GUI.** **If the unlocking mechanism is automatically applied after a specified time period, then the user account will be unlocked when the specified number of seconds have elapsed since the locking mechanism was engaged.** **If the lockout attempts is set to, for example, 5 attempts, then the user will be locked out after the 5th consecutive failed login attempt. This means that the 6th and subsequent attempts will fail to gain access to the TOE even if the credential being offered is correct.**

The evaluator shall examine the TSS to confirm that the TOE ensures that authentication failures by remote administrators cannot lead to a situation where no administrator access is available, either permanently or temporarily (e.g. by providing local logon which is not subject to blocking).

[4] VX series models don't support Web UI Fetaure

Evaluator Findings:
The evaluator shall examine the TSS to confirm that the TOE ensures that authentication failures by remote administrators cannot lead to a situation where no administrator access is available, either permanently or temporarily (e.g. by providing local logon which is not subject to blocking). The relevant information is found in the following section(s): TOE Summary Specification **FIA_AFL.1.** Upon investigation, the evaluator found that the TSS states that: **The failed authentication lockout does not apply to the local console, ensuring administrative access is always available.**

Verdict:
PASS.

5.1.3.1.2 FIA_AFL.1 AGD

The evaluator shall examine the guidance documentation to ensure that instructions for configuring the number of successive unsuccessful authentication attempts and time period (if implemented) are provided, and that the process of allowing the remote administrator to once again successfully log on is described for each “action” specified (if that option is chosen). If different actions or mechanisms are implemented depending on the secure protocol employed (e.g., TLS vs. SSH), all must be described.

Evaluator Findings:
The evaluator examined the guidance documentation to ensure that instructions for configuring the number of successive unsuccessful authentication attempts and time period (if implemented) are provided, and that the process of allowing the remote administrator to once again successfully log on is described for each “action” specified (if that option is chosen). If different actions or mechanisms are implemented depending on the secure protocol employed (e.g., TLS vs. SSH), all must be described. The relevant information is found in the following section(s): **Authentication Failure Handling** Upon investigation, the evaluator found that the AGD states that: **The locking mechanism can be configured to remain locked until an administrator unlocks the account, or it can be configured to unlock after a specified period of time.** **To configure, it requires following commands:** **‘aaa authentication attempts lockout unlock-time <time in seconds>’** Note: If the unlocking mechanism is automatically applied after a specified time period, then the user account will be unlocked when the specified number of seconds have elapsed since the locking mechanism was engaged. **‘aaa authentication attempts lockout max-fail <count>’** Note: The configurable range of failed attempts is between 1 to 15 attempts. **To unlock an account before lockout period elapses, following command is required:**

'aaa authentication attempts reset'

Note: The locking mechanisms apply to authentication attempts through both SSH and the GUI. The failed authentication lockout does not apply to the local console.

Note: The VX series models do not support the Web UI feature; therefore, the GUI/HTTPS logon method is not available on these models.

The evaluator shall examine the guidance documentation to confirm that it describes, and identifies the importance of, any actions that are required in order to ensure that administrator access will always be maintained, even if remote administration is made permanently or temporarily unavailable due to blocking of accounts as a result of FIA_AFL.1.

Evaluator Findings:
The evaluator examined the guidance documentation to confirm that it describes, and identifies the importance of, any actions that are required in order to ensure that administrator access will always be maintained, even if remote administration is made permanently or temporarily unavailable due to blocking of accounts as a result of FIA_AFL.1. The relevant information is found in the following section(s): **Authentication Failure Handling** Upon investigation, the evaluator found that the AGD states that: **Locally connected administrators are not subject to the lockout.** **As the locking mechanisms apply to authentication attempts through both SSH and the GUI[5]. The failed authentication lockout does not apply to the local console, ensuring administrative access is always available.**

Verdict:
PASS.

5.1.3.2 FIA_PMG_EXT.1 PASSWORD MANAGEMENT

5.1.3.2.1 FIA_PMG_EXT.1 TSS **[TD0792]**

The evaluator shall check that the TSS lists the supported special character(s) for the composition of administrator passwords.

Evaluator Findings:
The evaluated examined the TSS and verified that it lists the supported special character(s) for the composition of administrator passwords. The relevant information is found in the following section(s): TOE Summary Specification **FIA_PMG_EXT.1**.

[5] VX series models don't support Web UI Fetaure

Upon investigation, the evaluator found that the TSS states that: **The passwords can be composed of any combination of upper and lower case letters, numbers, and special characters (that include: "!", "@", "#", "$", "%", "^", "&", "*", "(", ")", "", "+", "-", ".", "/", ":", ";", "<", "=", ">", "?", "[", "\", "]", "^", "_", "`", "{", "|", "}", and "~".**

The evaluator shall check the TSS to ensure that the minimum_password_length parameter is configurable by a Security Administrator.

Evaluator Findings:
The evaluated examined the TSS and verified that the minimum_password_length parameter is configurable by a Security Administrator. The relevant information is found in the following section(s): TOE Summary Specification **FIA_PMG_EXT.1**. Upon investigation, the evaluator found that the TSS states that: **The minimum password length is settable by the Authorized Administrator and can range from 15 to 32 characters.**

The evaluator shall check that the TSS lists the range of values supported for the minimum_password_length parameter. The listed range shall include the value of 15.

Evaluator Findings:
The evaluated examined the TSS and verified that it lists the range of values supported for the minimum_password_length parameter. The listed range includes the value of 15. The relevant information is found in the following section(s): TOE Summary Specification **FIA_PMG_EXT.1**. Upon investigation, the evaluator found that the TSS states that: **The minimum password length is settable by the Authorized Administrator and can range from 15 to 32 characters.**

Verdict:
PASS.

5.1.3.2.2 FIA_PMG_EXT.1 AGD

The evaluator shall examine the guidance documentation to determine that it:

a) identifies the characters that may be used in passwords and provides guidance to security administrators on the composition of strong passwords, and

b) provides instructions on setting the minimum password length and describes the valid minimum password lengths supported.

Evaluator Findings:
The evaluator examined the guidance documentation to determine that it: a) identifies the characters that may be used in passwords and provides guidance to security administrators on the composition of strong passwords, and b) provides instructions on setting the minimum password length and describes the valid minimum password lengths supported. The relevant information is found in the following section(s): **Password Management** Upon investigation, the evaluator found that the AGD states that: **Passwords can be composed of any combination of upper and lower case letters, numbers, and the following special characters that include: [“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, “)”, [“”[1], “+”, “-“, “.”, “/”, “:”, “;”, “<”, “=”, “>”, “?”, “\”, “[“, “]”[2], “^”, “_”[3], “`”[4], “{“, “|”[5], “}”, and “~”]].** On the composition of strong passwords, the evaluator found that the AGD states: **The TOE can configure strong passwords, such as those with at least 15 characters long and the following complexity rules:** • **At least one uppercase letter** • **At least one lowercase letter** • **At least one number** • **At least one special character** **The appliance maintains a minimum password length of 8 characters by default. The minimum password length can be configured using:** **‘aaa authentication password local length minimum <count>’** **Note: It has a range of 8 to 32 characters. In CC mode of operation, the minimum length is 15 characters.**

Verdict:
PASS.

5.1.3.3 FIA_UIA_EXT.1 USER IDENTIFICATION AND AUTHENTICATION

5.1.3.3.1 FIA_UIA_EXT.1 TSS

The evaluator shall examine the TSS to determine that it describes the logon process for each logon method (local, remote (HTTPS, SSH, etc.)) supported for the product. This description shall contain information pertaining to the credentials allowed/used, any protocol transactions that take place, and what constitutes a “successful logon”.

Evaluator Findings:
The evaluator examined the TSS row **FIA_UIA_EXT.1** to determine that it describes the logon process for remote authentication mechanism (e.g. SSH public key, Web GUI password, etc.) and optional local authentication mechanisms supported by the TOE. This description shall contain information pertaining to the credentials allowed/used, any protocol transactions that take place, and what constitutes a “successful logon”.

The relevant information is found in the following section(s): TOE Summary Specification, **FIA_UIA_EXT.1**.

Upon investigation, the evaluator found that the TSS states that**: The TOE requires all users to be successfully identified and authenticated before allowing any TSF mediated actions to be performed. Administrative access to the TOE is facilitated through one of several interfaces:**

- **Directly connecting to each TOE appliance**
- **Remotely connecting to each appliance via SSHv2**
- **Remotely connecting to appliance GUI[6] via HTTPS/TLS**

Regardless of the interface at which the administrator interacts, the TOE prompts the user for a credential. Only after the administrative user presents the correct authentication credentials will they be granted access to the TOE administrative functionality. No TOE administrative access is permitted until an administrator is successfully identified and authenticated.

The TOE provides a local password-based authentication mechanism. The process for authentication is the same for administrative access whether administration is occurring via direct connection or remotely. At initial login, the administrative user is prompted to provide a username. After the user provides the username, the user is prompted to provide the administrative credential associated with the user account (e.g., password or SSH public/private key response). The TOE then either grants administrative access (if the combination of username and credential is correct) or indicates that the login was unsuccessful. The TOE does not provide a reason for failure in the cases of a login failure.

The evaluator shall examine the TSS to determine that it describes which actions are allowed before user identification and authentication. The description shall cover authentication and identification for local and remote TOE administration.

Evaluator Findings:
The evaluator examined the TSS row **FIA_UIA_EXT.1** and determined that it describes which actions are allowed before administrator identification and authentication. The description shall cover authentication and identification for local and remote TOE administration. The relevant information is found in the following section(s): TOE Summary Specification **FIA_UIA_EXT.1**. Upon investigation, the evaluator found that the TSS states that: **The TOE does not permit any administrative function to be accessible until after an administrator is successfully identified and**

[6] VX series models don't support Web UI Fetaure

authenticated, but the TOE does display the warning banner prior to requiring user identification and authentication.

Upon investigation, the evaluator also found that: **The TOE provides a local password-based authentication mechanism.**
The process for authentication is the same for administrative access whether administration is occurring via direct connection or remotely[7]. At initial login, the administrative user is prompted to provide a username. After the user provides the username, the user is prompted to provide the administrative credential associated with the user account (e.g., password or SSH public/private key response).

For distributed TOEs the evaluator shall examine that the TSS details how Security Administrators are authenticated and identified by all TOE components. If not, all TOE components support authentication of Security Administrators according to FIA_UIA_EXT.1 and FIA_UAU_EXT.2, the TSS shall describe how the overall TOE functionality is split between TOE components including how it is ensured that no unauthorized access to any TOE component can occur.

Evaluator Findings:
The TOE is not a distributed TOE hence this assurance activity is not applicable.

For distributed TOEs, the evaluator shall examine the TSS to determine that it describes for each TOE component which actions are allowed before user identification and authentication. The description shall cover authentication and identification for local and remote TOE administration. For each TOE component that does not support authentication of Security Administrators according to FIA_UIA_EXT.1 and FIA_UAU_EXT.2 the TSS shall describe any unauthenticated services/services that are supported by the component.

Evaluator Findings:
The TOE is not a distributed TOE hence this assurance activity is not applicable.

Verdict:
PASS.

5.1.3.3.2 FIA_UIA_EXT.1 AGD

The evaluator shall examine the guidance documentation to determine that any necessary preparatory steps (e.g., establishing credential material such as pre- shared keys, tunnels, certificates, etc.) to logging in are described. For each supported the login method, the evaluator shall ensure the guidance documentation provides clear instructions for successfully logging on. If configuration is necessary to ensure the services provided before login are limited, the evaluator shall determine that the guidance documentation provides sufficient instruction on limiting the allowed services.

[7] VX series models don't support Web UI feataure

Evaluator Findings:

The evaluator examined the guidance documentation and determined that any necessary preparatory steps (e.g., establishing credential material such as pre- shared keys, tunnels, certificates, etc.) to logging in are described. For each supported the login method, the evaluator ensured the guidance documentation provides clear instructions for successfully logging on. If configuration is necessary to ensure the services provided before login are limited, the evaluator determined that the guidance documentation provides sufficient instruction on limiting the allowed services.

The relevant information is found in the following section:

Sections	Preparatory steps
Using the Console	To access the CLI of the appliance using the console port.
Connect to Appliance via SSH	To access the CLI of the appliance using the SSH.
Connect to Appliance via WEB UI[8]	To manage TOE using WEB UI which is available after the initial setup through the serial console.
User Creation via the CLI	To configure users and their roles.
User Creation via the Web UI	
Login Banners	To customize banners

Upon investigation, the evaluator found the information in AGD and summarized as:

Regardless of method of administering the TOE, the user is presented with an authentication prompt. At the authentication prompt the username of the administrator and credential (either password or SSH key) must be presented. Administration is available only after the correct username/credential combination is presented.

The section '**Login Banners**' describes customization of banners as follows:

To configure the messages which users see when they log in to the appliance:

- **To change the local login message only, use the following command:**

  'hostname (config) # banner login-local "<text>"'
- **To change the remote login message only, use the following command:**

  'hostname (config) # banner login-remote "<text>"'
- **To change the message of the day, use the following command:**

  'hostname (config) # banner motd "<text>"'

The section '**Login Banners**' also describes that display of the Login banner is the only service that is available prior to identification and authentication. No configuration is required to ensure that the access to services is limited prior to login.

[8] VX series models don't support Web UI feature.

Verdict:
PASS.

5.1.3.4 FIA_UAU_EXT.2 PASSWORD-BASED AUTHENTICATION MECHANISM

Evaluation Activities for this requirement are covered under those for FIA_UIA_EXT.1. If other authentication mechanisms are specified, the evaluator shall include those methods in the activities for FIA_UIA_EXT.1.

5.1.3.5 FIA_UAU.7 PROTECTED AUTHENTICATION FEEDBACK

5.1.3.5.1 FIA_UAU.7 TSS

None.

5.1.3.5.2 FIA_UAU.7 AGD

The evaluator shall examine the guidance documentation to determine that any necessary preparatory steps to ensure authentication data is not revealed while entering for each local login allowed.

Evaluator Findings:
The evaluator examined the guidance documentation to determine that any necessary preparatory steps to ensure authentication data is not revealed while entering for each local login allowed. The relevant information is found in the following section(s): **Protection Authentication Feedback** Upon investigation, the evaluator found that the AGD states that: **The TOE does not provide any feedback for the password characters entered. This is by default and does not require any configuration.**

Verdict:
PASS.

5.1.4 SECURITY MANAGEMENT (FMT)

5.1.4.1 FMT_MOF.1/MANUALUPDATE

5.1.4.1.1 FMT_MOF.1/MANUALUPDATE TSS

For distributed TOEs see chapter 2.4.1.1. There are no specific requirements for non-distributed TOEs.

Evaluator Findings:
The TOE is not a distributed TOE and there are no specific requirements for non-distributed TOES; hence, this assurance activity is not applicable.

Verdict:
PASS.

5.1.4.1.2 FMT_MOF.1/MANUALUPDATE AGD

The evaluator shall examine the guidance documentation to determine that any necessary steps to perform manual update are described. The guidance documentation shall also provide warnings regarding functions that may cease to operate during the update (if applicable).

Evaluator Findings:
The evaluator examined the guidance documentation and determined that any necessary steps to perform manual update are described. The guidance documentation also provides warnings regarding functions that may cease to operate during the update (if applicable). The relevant information is found in the following section(s): **Software Updates** Upon investigation, the evaluator found that the AGD states: **To perform a software update, following command is required:** • Download the software image: **'hostname (config) # image fetch <location of image>'** • Install the downloaded software image: **'hostname (config) # image install <image-lms_7.9.0.img>'** **'hostname (config) # image boot next'** Additionally, the evaluator found that the AGD states that: **No functionality will cease during the update process. Device will remain fully operational until the administrator reboots the product.**

For distributed TOEs the guidance documentation shall describe all steps how to update all TOE components. This shall contain description of the order in which components need to be updated if the order is relevant to the update process. The guidance documentation shall also provide warnings regarding functions of TOE components and the overall TOE that may cease to operate during the update (if applicable).

Evaluator Findings:
The TOE is not a distributed TOE; hence, this assurance activity is not applicable.

Verdict:
PASS.

5.1.4.2 FMT_MTD.1/COREDATA MANAGEMENT OF TSF DATA

5.1.4.2.1 FMT_MTD.1/COREDATA TSS

The evaluator shall examine the TSS to determine that, for each administrative function identified in the guidance documentation; those that are accessible through an interface prior to administrator log-in are identified. For each of these functions, the evaluator shall also confirm that the TSS details how the ability to manipulate the TSF data through these interfaces is disallowed for non-administrative users.

Evaluator Findings:
The evaluator confirmed that the TSS row **FMT_MTD.1/CoreData** details how the ability to manipulate the TSF data through these interfaces is disallowed for non-administrative users. The relevant information is found in the following section(s): TOE Summary Specification **FMT_MTD.1/CoreData**. Upon investigation, the evaluator found that the TSS states that: **The only access the TOE allows prior to the successful identification and authentication of a user is the access banner displayed at each login prompt. No other security functions are accessible.** **The TOE implements role-based access control to manipulate the TSF data. Administrative users are required to login before being provided with access to any administrative functions.** **The TOE performs role-based authorization, using TOE platform authorization mechanisms, to grant access to all the privileged levels.**

If the TOE supports handling of X.509v3 certificates and implements a trust store, the evaluator shall examine the TSS to determine that it contains sufficient information to describe how the ability to manage the TOE's trust store is restricted.

Evaluator Findings:
The TOE supports handling of X.509v3 certificates and implements a trust store, the evaluator examined the TSS row **FMT_MTD.1/CoreData** and determined that it contains sufficient information to describe how the ability to manage the TOE's trust store is restricted. The relevant information is found in the following section(s): TOE Summary Specification **FMT_MTD.1/CoreData**. Upon investigation, the evaluator found that the TSS states that: **The TOE performs role-based authorization, using TOE platform authorization mechanisms, to grant access to all the privileged levels.** **The term "Security Administrator" is used in this ST to refer to any user which has been assigned a role that is permitted to perform the relevant action; therefore, has the appropriate privileges to perform the requested functions. Users without the appropriate privilege level do not have access to TOE functionality including administration of X.509 certificates via TOE's trust store.**

Verdict:
PASS.

5.1.4.2.2 FMT_MTD.1/COREDATA AGD

The evaluator shall review the guidance documentation to determine that each of the TSF-data-manipulating functions implemented in response to the requirements of the c PP is identified, and that configuration information is provided to ensure that only administrators have access to the functions.

Evaluator Findings:
The evaluator reviewed the guidance documentation and determined that each of the TSF-data-manipulating functions implemented in response to the requirements of the cPP is identified, and that configuration information is provided to ensure that only administrators have access to the functions. The relevant information is found in the following section(s): **TOE Administration, Setting the Time, Software Integrity, Software Updates, Enabling CC-NDcPP Compliance Mode, Using an Audit Server** and **Automatic Logout due to Inactivity** Upon investigation, the evaluator found that the AGD states that: **Only authorized administrators can update and modify TOE functions.** Additionally, the configuration available for each of the data manipulating functions available on the TOE are described in AGD, it is consistent with ST.

If the TOE supports handling of X.509v3 certificates and provides a trust store, the evaluator shall review the guidance documentation to determine that it provides sufficient information for the administrator to configure and maintain the trust store in a secure way. If the TOE supports loading of CA certificates, the evaluator shall review the guidance documentation to determine that it provides sufficient information for the administrator to securely load CA certificates into the trust store. The evaluator shall also review the guidance documentation to determine that it explains how to designate a CA certificate a trust anchor.

Evaluator Findings:
If the TOE supports handling of X.509v3 certificates and provides a trust store, the evaluator reviewed the guidance documentation and determined that it provides sufficient information for the administrator to configure and maintain the trust store in a secure way. If the TOE supports loading of CA certificates, the evaluator reviewed the guidance documentation and determined that it provides sufficient information for the administrator to securely load CA certificates into the trust store. The evaluator also reviewed the guidance documentation and determined that it explains how to designate a CA certificate a trust anchor. The relevant information is found in the following section(s): **Adding to certificates in Trust Store** Upon investigation, the evaluator found that the AGD **provides specific instructions for adding certificates in the TOE Trust Store via CLI as well as via Web UI.** To add certificates via Web UI, following instructions are required:

- **On the Web UI[9], select Settings Tab**
- **Select Certificates/Keys**
- **Click Add Root/Intermediate CA Certificate**
- **Choose file then commit**

To add certificates via CLI, following commands are required:

'crypto certificate name xxx public-cert pem xxx'

'crypto certificate ca-list default-ca-list name xxx'

This process is for adding either an intermediate or root certificate to the trust store.

Verdict:
PASS.

5.1.4.3 FMT_SMF.1 SPECIFICATION OF MANAGEMENT FUNCTIONS

The security management functions for FMT_SMF.1 are distributed throughout the cPP and are included as part of the requirements in FTA_SSL_EXT.1, FTA_SSL.3, FTA_TAB.1, FMT_MOF.1/ManualUpdate, FMT_MOF.1/AutoUpdate (if included in the ST), FIA_AFL.1, FIA_X509_EXT.2.2 (if included in the ST), FPT_TUD_EXT.1.2 & FPT_TUD_EXT.2.2 (if included in the ST and if they include an administrator-configurable action), FMT_MOF.1/Services, and FMT_MOF.1/Functions (for all of these SFRs that are included in the ST), FMT_MTD, FPT_TST_EXT, and any cryptographic management functions specified in the reference standards. Compliance to these requirements satisfies compliance with FMT_SMF.1.

5.1.4.3.1 FMT_SMF.1 TSS (CONTAINING ALSO REQUIREMENTS ON GUIDANCE DOCUMENTATION AND TESTS)

The evaluator shall examine the TSS, Guidance Documentation and the TOE as observed during all other testing and shall confirm that the management functions specified in FMT_SMF.1 are provided by the TOE.

Evaluator Findings:
The evaluator examined the TSS row **FMT_SMF.1**, the Guidance Documentation and the TOE as observed during all other testing and confirmed that the management functions specified in FMT_SMF.1 are provided by the TOE. The relevant information is found in the following section(s): TOE Summary Specification **FMT_SMF.1** and **TOE administration** in the Guidance document**.**

[9] VX series models don't support Web UI Fetaure

Upon investigation, the evaluator found that the TSS states that: **The specific management capabilities include:**

- **Ability to administer the TOE locally**
- **Ability to administer the TOE remotely**[10]
- **Ability to configure the access banner**
- **Ability to configure the session inactivity time before session termination**
- **Ability to update the TOE, and to verify the updates using digital signature capability prior to installing those updates**
- **Ability to configure the authentication failure parameters**
- **Ability to modify the behavior of the transmission of audit data to an external IT entity and the handling of local audit data**
- **Ability to configure the cryptographic functionality**
- **Ability to re-enable an Administrator account**
- **Ability to set the time which is used for time-stamps**
- **Ability to configure NTP**
- **Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors**
- **Ability to import X.509v3 certificates to the TOE's trust store**
- **Ability to manage the trusted public keys database**

The evaluator also found that the AGD describes **all the details for the management functions specified in FMT_SMF.1 and found that the administrative activities are consistent with the TSS.**

The evaluator shall confirm that the TSS details which security management functions are available through which interface(s) (local administration interface, remote administration interface).

Evaluator Findings:
The evaluator confirmed that the TSS row **FMT_SMF.1** details which security management functions are available through which interface(s) (local administration interface, remote administration interface). The relevant information is found in the following section(s): TOE Summary Specification **FMT_SMF.1**

[10] VX series models don't support Web UI Fetaure

Upon investigation, the evaluator found that the TSS states that: **The specific management capabilities include:**

- **Ability to administer the TOE locally (CLI);**
- **Ability to administer the TOE remotely (GUI[11] & CLI);**
- **Ability to configure the access banner (GUI & CLI);**
- **Ability to configure the session inactivity time before session termination (CLI);**
- **Ability to update the TOE, and to verify the updates using digital signature capability prior to installing those updates (CLI);**
- **Ability to configure the authentication failure parameters (CLI);**
- **Ability to modify the behavior of the transmission of audit data to an external IT entity and the handling of local audit data (CLI);**
- **Ability to configure the cryptographic functionality (CLI);**
- **Ability to re-enable an Administrator account (CLI);**
- **Ability to set the time which is used for time-stamps (GUI & CLI);**
- **Ability to configure NTP (CLI);**
- **Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors (GUI & CLI);**
- **Ability to import X.509v3 certificates to the TOE's trust store (GUI & CLI)**
- **Ability to manage the trusted public keys database (CLI).**

The evaluator shall examine the TSS and Guidance Documentation to verify they both describe the local administrative interface.

Evaluator Findings:
The evaluator examined the TSS row **FMT_SMF.1** and the Guidance Documentation to verify they both describe the local administrative interface. The relevant information is found in the following section(s): TOE Summary Specification **FMT_SMF.1** and **Initial Setup of the TOE** Upon investigation, the evaluator found that the TSS states that: **The TOE may be managed via the CLI (console & SSH) or GUI (HTTPS).** Upon investigation, the evaluator also found that the AGD describes **the local interface and the configurations required to communicate on the interface.**

The evaluator shall ensure the Guidance Documentation includes appropriate warnings for the administrator to ensure the interface is local.

Evaluator Findings:
The evaluator ensured the Guidance Documentation includes appropriate warnings for the administrator to ensure the interface is local.

[11] VX series models don't support Web UI Fetaure

The relevant information is found in the following section(s): **Initial Setup of the TOE**

Upon investigation, the evaluator found that the AGD describes: **The steps associated with connecting to the serial port of a computer. This sufficiently ensures that the interface is a local interface.**

For distributed TOEs with the option 'ability to configure the interaction between TOE components' the evaluator shall examine that the ways to configure the interaction between TOE components is detailed in the TSS and Guidance Documentation.

Evaluator Findings:
The TOE is not a distributed TOE hence this assurance activity is not applicable.

The evaluator shall check that the TOE behaviour observed during testing of the configured SFRs is as described in the TSS and Guidance Documentation.

Evaluator Findings:

The evaluator checked that the TOE behaviour observed during testing of the configured SFRs is as described in the TSS row **FMT_SMF.1** and the Guidance Documentation below mentioned sections.

TSS	AGD Sections	Test Cases
Ability to administer the TOE locally (CLI);	Using the Console	FIA_UIA_EXT.1 Test #1
Ability to administer the TOE remotely (GUI[12] & CLI);	Connect to Appliance via SSH and Connect to Appliance via WEB UI	FIA_UIA_EXT.1 Test #1
Ability to configure the access banner (GUI & CLI);	Login Banners	FTA_TAB.1 Test #1
Ability to configure the session inactivity time before session termination (CLI);	Automatic Logout due to Inactivity	FTA_SSL_EXT.1.1 Test #1, FTA_SSL.3 test #1
Ability to update the TOE, and to verify the updates using digital signature capability prior to installing those updates (CLI);	Software Updates	FPT_TUD_EXT.1Test #1, FMT_MOF.1/ManualUpdate Test #2
Ability to configure the authentication failure parameters (CLI);	Authentication Failure Handling	FIA_AFL.1 Test #1 and FIA_AFL.1 Test#2b
Ability to modify the behavior of the transmission of audit data to an external IT entity	Using an Audit Server	FAU_STG_EXT.1 Test#1

[12] VX series models don't support Web UI Fetaure

and the handling of local audit data (CLI);		
Ability to configure the cryptographic functionality (CLI);	Configuring X.509 certificate Authentication for the Web UI	FIA_X509_EXT.3 Test #1
Ability to re-enable an Administrator account (CLI);	Authentication Failure Handling	FIA_AFL.1 Test #2a
Ability to set the time which is used for time-stamps (GUI[13] & CLI);	Setting Time	FPT_STM_EXT.1 test #1
Ability to configure NTP (CLI);	Setting Time	FCS_NTP_EXT.1.1 Test #1 and FCS_NTP_EXT.1.2Test #1
Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors (GUI & CLI);	Addition and Removal of Certificates from Trust Store and Configuring X.509 certificate Authentication for the Web UI	FIA_X509_EXT.1/ Rev Test #1b
Ability to import X.509v3 certificates to the TOE's trust store (GUI & CLI)	Addition of Certificates to Trust Store	FIA_X509_EXT.1/ Rev Test #1a
Ability to manage the trusted public keys database (CLI).	Configuring SSH Public Keys	FCS_SSHS_EXT.1.2 Test #1

Verdict:
PASS.

5.1.4.3.2 FMT_SMF.1 AGD

Evaluator Findings:
See section 5.1.4.3.1 of this document for AGD activities.

Verdict:
PASS.

5.1.4.4 FMT_SMR.2 RESTRICTIONS ON SECURITY ROLES

5.1.4.4.1 FMT_SMR.2 TSS

The evaluator shall examine the TSS to determine that it details the TOE supported roles and any restrictions of the roles involving administration of the TOE.

Evaluator Findings:
The evaluator examined the TSS row **FMT_SMR.2** and **FMT_MTD.1/CoreData** and determined that it details the TOE supported roles and any restrictions of the roles involving administration of the TOE.

[13] VX series models don't support Web UI Fetaure

The relevant information is found in the following section(s): TOE Summary Specification **FMT_SMR.2** and **FMT_MTD.1/CoreData**.

Upon investigation, the evaluator found that the TSS states that **The TOE supports several types of administrative user roles. Collectively these roles comprise the Security Administrator.**

The supported roles include:

- **Admin: The system administrator is a "super user" who has all capabilities. The primary function of this role is to configure the system.**
- **Monitor: The system monitor has read-only access to some things the admin role can change or configure.**
- **Operator: The system operator has a subset of the capabilities associated with the admin role. Its primary function is configuring and monitoring the system.**
- **Analyst: The system analyst focuses on data plane analysis and possesses several capabilities, including setting up alerts and reports.**
- **Auditor: The system auditor reviews audit logs and performs forensic analysis to trace how events occurred.**

Each of the predefined administrative roles have a set of permissions that will grant them access to the TOE data, though with some roles, the access is limited.

The TOE performs role-based authorization, using TOE platform authorization mechanisms, to grant access to all the privileged levels.

Verdict:
PASS.

5.1.4.4.2 FMT_SMR.2 AGD

The evaluator shall review the guidance documentation to ensure that it contains instructions for administering the TOE both locally and remotely, including any configuration that needs to be performed on the client for remote administration.

Evaluator Findings:

The evaluator reviewed the AGD and ensured that it contains instructions for administering the TOE both locally and remotely, including any configuration that needs to be performed on the client for remote administration.

The relevant information is found in the following section(s):
Connect to Appliance via SSH, Connect to Appliance via WEB UI and **Enabling CC-NDcPP Compliance Mode**

Upon investigation, the evaluator found that the AGD states that:

Instructions for administering the CLI of the appliance using the SSH, follow these steps:

- **Open a terminal program on your system, such as Putty.**
- **Enter appliance IP address i.e. IP ass assigned to the ether1.**

Administering the WEB UI[14] is available after the initial setup through the serial console, following steps are required:

- **Launch a web browser from a laptop that is network-connected.**
- **Point the browser at the same IP address that was assigned to the ether1 followed by /login (for example, https://a.b.c.d/login).**
- **On the sign-in page, enter the administrator username and password. Then click Sign In.**

Additionally, the section titled '**Enabling CC-NDcPP Compliance Mode**' of AGD states that: **After compliance has been enabled on an appliance per the below instructions, you must use SSH from a server or desktop that has the proper ciphers**.

Also, **for each applicable function, the method for configuring the function via the Web UI (remote) and via the CLI (local/remote) is described.**

Verdict:
PASS.

5.1.5 PROTECTION OF THE TSF (FPT)

5.1.5.1 FPT_SKP_EXT.1 PROTECTION OF TSF DATA (FOR READING OF ALL PRE-SHARED, SYMMETRIC AND PRIVATE KEYS)

5.1.5.1.1 FPT_SKP_EXT.1 TSS

The evaluator shall examine the TSS to determine that it details how any pre- shared keys, symmetric keys, and private keys are stored and that they are unable to be viewed through an interface designed specifically for that purpose, as outlined in the application note. If these values are not stored in plaintext, the TSS shall describe how they are protected/obscured.

Evaluator Findings:
The evaluator examined the TSS row **FPT_SKP_EXT.1** and determined that it details how any pre-shared keys, symmetric keys, and private keys are stored and that they are unable to be viewed through an interface designed specifically for that purpose, as outlined in the application note. If these values are not stored in plaintext, the TSS describes how they are protected/obscured. The relevant information is found in the following section(s): TOE Summary Specification **FPT_SKP_EXT.1**.and section the **Cryptographic Key Destruction**.

[14] VX series models don't support Web UI Fetaure

Upon investigation, the evaluator found that the TSS states that: **The TOE stores all private keys in plaintext in a secure directory that is not readily accessible to administrators; hence no interface access.**

Additionally, the section titled **'Cryptographic Key Destruction'** describes the storage of each key.

Verdict:
PASS.

5.1.5.2 FPT_APW_EXT.1 PROTECTION OF ADMINISTRATOR PASSWORDS

5.1.5.2.1 FPT_APW_EXT.1 TSS

The evaluator shall examine the TSS to determine that it details all authentication data that are subject to this requirement, and the method used to obscure the plaintext password data when stored. The TSS shall also detail passwords are stored in such a way that they are unable to be viewed through an interface designed specifically for that purpose, as outlined in the application note.

Evaluator Findings:
The evaluator examined the TSS to determine that it details all authentication data that are subject to this requirement, and the method used to obscure the plaintext password data when stored. The TSS also detailed passwords are stored in such a way that they are unable to be viewed through an interface designed specifically for that purpose, as outlined in the application note. The relevant information is found in the following section(s): TOE Summary Specification **FPT_APW_EXT.1.** Upon investigation, the evaluator found that the TSS states that: **The TOE stores Security Administrator passwords. All passwords are stored in a secure directory that is not readily accessible to administrators. The passwords are stored SHA-512 hashed and not in plaintext.**

Verdict:
PASS.

5.1.5.3 FPT_TST_EXT.1 TSF TESTING

5.1.5.3.1 FPT_TST_EXT.1 TSS

The evaluator shall examine the TSS to ensure that it details the self-tests that are run by the TSF; this description should include an outline of what the tests are actually doing (e.g., rather than saying "memory is tested", a description similar to "memory is tested by writing a value to each memory location and reading it back to ensure it is identical to what was written" shall be used).

Evaluator Findings:
The evaluator examined the TSS row **FPT_TST_EXT.1** and ensured that it details the self-tests that are run by the TSF; this description should include an outline of what the tests are actually doing.

The relevant information is found in the following section(s): TOE Summary Specification **FPT_TST_EXT.1**.

Upon investigation, the evaluator found that the TSS states that: **The TOE runs a suite of self-tests during initial start-up to verify its correct operation. If any of the tests fail, the TOE will enter into an error state until an Administrator intervenes.**

During the system bootup process (power on or reboot), all the cryptographic modules perform the Cryptographic Power on Startup Test (POST).

The Cryptographic POST verifies that each cryptographic algorithm specified in FCS_COP.1 requirements is passing a Known Answer Test (KAT). The KAT demonstrates that the algorithm is functioning properly by invoking the algorithm with hard coded keys and messages and comparing the result to a pre-computed, known to be correct value. TOE performs Cryptographic POST that is indicated as 'FIPS crypto POST'.

The Software Integrity Test is run automatically on start-up, and whenever the system images are loaded. A hash verification is used to confirm the image file to be loaded has not been corrupted and has maintained its integrity.

The evaluator shall ensure that the TSS makes an argument that the tests are sufficient to demonstrate that the TSF is operating correctly.

Evaluator Findings:
The evaluator ensured that the TSS row **FPT_TST_EXT.1** makes an argument that the tests are sufficient to demonstrate that the TSF is operating correctly. The relevant information is found in the following section(s): TOE Summary Specification **FPT_TST_EXT.1**. Upon investigation, the evaluator found that the TSS states that: **The Software Integrity Test is run automatically on start-up, and whenever the system images are loaded. A hash verification is used to confirm the image file to be loaded has not been corrupted and has maintained its integrity.** **These tests are sufficient to verify that the correct version of the TOE software is running as well as that the cryptographic operations are all performing as expected. Both of these functions are required to ensure that the TOE is operating as expected and data that the user expects to be encrypted in not transferred in plaintext.**

For distributed TOEs the evaluator shall examine the TSS to ensure that it details which TOE component performs which self-tests and when these self-tests are run.

Evaluator Findings:
The TOE is not a distributed TOE hence this assurance activity is not applicable.

Verdict:
PASS.

5.1.5.3.2 FPT_TST_EXT.1 AGD

The evaluator shall also ensure that the guidance documentation describes the possible errors that may result from such tests, and actions the administrator should take in response; these possible errors shall correspond to those described in the TSS.

Evaluator Findings:
The evaluator also ensured that the guidance documentation describes the possible errors that may result from such tests, and actions the administrator should take in response; these possible errors correspond to those described in the TSS. The relevant information is found in the following section(s): **Cryptographic POST** and **Software Integrity** Upon investigation, the evaluator found the information in AGD and paraphrased it as: **If any of the tests fail, then the appliance enters failed state which is also described in the TSS.** Also, **No specific administrative interaction is required if an error is encountered. The reboot process will happen automatically and TOE will not start unless the tests have passed. Administrator should contact vendor support team in case of device stuck in boot loop.**

For distributed TOEs the evaluator shall ensure that the guidance documentation describes how to determine from an error message returned which TOE component has failed the self-test.

Evaluator Findings:
The TOE is not a distributed TOE hence this assurance activity is not applicable.

Verdict:
PASS.

5.1.5.4 FPT_TUD_EXT.1 TRUSTED UPDATE

5.1.5.4.1 FPT_TUD_EXT.1 TSS

The evaluator shall verify that the TSS describes how to query the currently active version. If a trusted update can be installed on the TOE with a delayed activation, the TSS needs to describe how and when the inactive version becomes active. The evaluator shall verify this description.

Evaluator Findings:
The evaluator verified that the TSS row **FPT_TUD_EXT.1** describes how to query the currently active version. If a trusted update can be installed on the TOE with a delayed activation, the TSS describes how and when the inactive version becomes active. The evaluator verified this description.

The relevant information is found in the following section(s): TOE Summary Specification **FPT_TUD_EXT.1**.

Upon investigation, the evaluator found that the TSS states that: **The Security Administrator can query the software version running on the TOE and the most recently downloaded software version so the TOE does support delayed activation.**

Following successful authentication authorized administrators can perform management actions such as query the current version of the TOE software using CLI commands 'show version'.

An image that passes an integrity check will be installed. The new image remains inactive until the TOE is rebooted to the new image. Installed image integrity is further verified against tampering before the new image is allowed to become active on reboot, and failure will revert to the previous valid image.

The evaluator shall verify that the TSS describes all TSF software update mechanisms for updating the system firmware and software (for simplicity the term 'software' will be used in the following although the requirements apply to firmware and software).

Evaluator Findings:

The evaluator verified that the TSS describes all TSF software update mechanisms for updating the system firmware and software.

The relevant information is found in the following section(s): TOE Summary Specification **FPT_TUD_EXT.1**.

Upon investigation, the evaluator found that the TSS states that: **When software updates are made available by FireEye, the Security Administrator can download them from authorized website, and install them manually, at which time the system first verifies the integrity of the downloaded image before installing. No other update mechanism is available.**

Software updates are downloaded to the TOE via an 'image fetch' command on the CLI. Software images will not be installed without explicit administrative intervention.

The evaluator shall verify that the description includes a digital signature verification of the software before installation and that installation fails if the verification fails. Alternatively, an approach using a published hash can be used. In this case the TSS shall detail this mechanism instead of the digital signature verification mechanism.

Evaluator Findings:

The evaluator verified that the description includes a digital signature verification of the software before installation and that installation fails if the verification fails. Alternatively, an approach using a published hash can be used. In this case the TSS details this mechanism instead of the digital signature verification mechanism.

The relevant information is found in the following section(s): TOE Summary Specification **FPT_TUD_EXT.1**.

Upon investigation, the evaluator found that the TSS states that: **When software updates are made available by FireEye, the Security Administrator can download them from authorized website, and install them manually, at which time the system first verifies the integrity of the downloaded image before installing. No other update mechanism is available.**

The TOE image files are digitally signed (2048-bit RSA/SHA-256) by the vendor, so their integrity can be verified during the upgrade process.

An image that fails an integrity check will not be installed. An image that passes an integrity check will be installed. The new image remains inactive until the TOE is rebooted to the new image. Installed image integrity is further verified against tampering before the new image is allowed to become active on reboot, and failure will revert to the previous valid image.

The evaluator shall verify that the TSS describes the method by which the digital signature or published hash is verified to include how the candidate updates are obtained, the processing associated with verifying the digital signature or published hash of the update, and the actions that take place for both successful and unsuccessful signature verification or published hash verification.

Evaluator Findings:

The evaluator verified that the TSS describes the method by which the digital signature or published hash is verified to include how the candidate updates are obtained, the processing associated with verifying the digital signature or published hash of the update, and the actions that take place for both successful and unsuccessful signature verification or published hash verification.

The relevant information is found in the following section(s): TOE Summary Specification **FPT_TUD_EXT.1**.

Upon investigation, the evaluator found that the TSS states that: **When software updates are made available by FireEye, the Security Administrator can download them from authorized website, and install them manually, at which time the system first verifies the integrity of the downloaded image before installing. No other update mechanism is available.**

The TOE image files are digitally signed (2048-bit RSA/SHA-256) by the vendor, so their integrity can be verified during the upgrade process.

An image that fails an integrity check will not be installed. An image that passes an integrity check will be installed. The new image remains inactive until the TOE is rebooted to the new image. Installed image integrity is further verified against tampering before the new image is allowed to become active on reboot, and failure will revert to the previous valid image.

If the options ‘support automatic checking for updates’ or ‘support automatic updates’ are chosen from the selection in FPT_TUD_EXT.1.2, the evaluator shall verify that the TSS explains what actions are involved in automatic checking or automatic updating by the TOE, respectively.

Evaluator Findings:
The options ‘support automatic checking for updates’ or ‘support automatic updates’ are not selected in the ST hence this assurance activity is not applicable.

For distributed TOEs, the evaluator shall examine the TSS to ensure that it describes how all TOE components are updated, that it describes all mechanisms that support continuous proper functioning of the TOE during update (when applying updates separately to individual TOE components) and how verification of the signature or checksum is performed for each TOE component. Alternatively, this description can be provided in the guidance documentation. In that case the evaluator should examine the guidance documentation instead.

Evaluator Findings:
The TOE is not a distributed TOE hence this assurance activity is not applicable.

If a published hash is used to protect the trusted update mechanism, then the evaluator shall verify that the trusted update mechanism does involve an active authorization step of the Security Administrator, and that download of the published hash value, hash comparison and update is not a fully automated process involving no active authorization by the Security Administrator. In particular, authentication as Security Administration according to FMT_MOF.1/ManualUpdate needs to be part of the update process when using published hashes.

Evaluator Findings:
ST does not claim ‘Published hash’ hence this assurance activity is not applicable.

Verdict:
PASS.

5.1.5.4.2 FPT_TUD_EXT.1 AGD

The evaluator shall verify that the guidance documentation describes how to query the currently active version. If a trusted update can be installed on the TOE with a delayed activation, the guidance documentation needs to describe how to query the loaded but inactive version.

Evaluator Findings:
The evaluator verified that the guidance documentation describes how to query the currently active version. If a trusted update can be installed on the TOE with a delayed activation, the guidance documentation describes how to query the loaded but inactive version. The relevant information is found in the following section(s): **Software Updates** Upon investigation, the evaluator found that the AGD states that:

The Security Administrator can query the software version running on the TOE and the most recently downloaded software version, so the TOE does support delayed activation.

Following command query the currently active version and view installation status which allows the administrator to see the installed but inactive version:

'show images'

The evaluator shall verify that the guidance documentation describes how the verification of the authenticity of the update is performed (digital signature verification or verification of published hash). The description shall include the procedures for successful and unsuccessful verification. The description shall correspond to the description in the TSS.

Evaluator Findings:

The evaluator verified that the guidance documentation **section 10** describes how the verification of the authenticity of the update is performed (digital signature verification). The description includes the procedures for successful and unsuccessful verification. The description corresponds to the description in the TSS.

The relevant information is found in the following section(s): **Software Updates**

Upon investigation, the evaluator found that the AGD states that:

Software image files are digitally signed so their integrity can be automatically verified during the upgrade process. An image that fails an integrity check will not be loaded; it is consistent with the TSS.

If a published hash is used to protect the trusted update mechanism, the evaluator shall verify that the guidance documentation describes how the Security Administrator can obtain authentic published hash values for the updates.

Evaluator Findings:

ST does not claim 'Published hash' hence this assurance activity is not applicable.

For distributed TOEs the evaluator shall verify that the guidance documentation describes how the versions of individual TOE components are determined for FPT_TUD_EXT.1, how all TOE components are updated, and the error conditions that may arise from checking or applying the update (e.g. failure of signature verification, or exceeding available storage space) along with appropriate recovery actions. . The guidance documentation only has to describe the procedures relevant for the Security Administrator; it does not need to give information about the internal communication that takes place when applying updates.

Evaluator Findings:

The TOE is not a distributed TOE hence this assurance activity is not applicable.

If this was information was not provided in the TSS: For distributed TOEs, the evaluator shall examine the Guidance Documentation to ensure that it describes how all TOE components are updated, that it describes all mechanisms that support continuous proper functioning of the TOE during update (when applying updates separately to individual TOE components) and how verification of the signature or checksum is performed for each TOE component.

Evaluator Findings:
The TOE is not a distributed TOE hence this assurance activity is not applicable.

If this was information was not provided in the TSS: If the ST author indicates that a certificate-based mechanism is used for software update digital signature verification, the evaluator shall verify that the Guidance Documentation contains a description of how the certificates are contained on the device. The evaluator also ensures that the Guidance Documentation describes how the certificates are installed/updated/selected, if necessary.

Evaluator Findings:
Certificate-based mechanism is not used for software update digital signature verification hence this assurance activity is not applicable.

Verdict:
PASS.

5.1.5.5 FPT_STM_EXT.1 RELIABLE TIME STAMPS

5.1.5.5.1 FPT_STM_EXT.1 TSS **[TD0632]**

The evaluator shall examine the TSS to ensure that it lists each security function that makes use of time, and that it provides a description of how the time is maintained and considered reliable in the context of each of the time related functions.

Evaluator Findings:
The evaluator examined the TSS row **FPT_STM_EXT.1** and ensured that it lists each security function that makes use of time, and that it provides a description of how the time is maintained and considered reliable in the context of each of the time related functions. The relevant information is found in the following section(s): TOE Summary Specification **FPT_STM_EXT.1.** Upon investigation, the evaluator found that the TSS states that: **The clock function is reliant on the system clock provided by the underlying hardware.** **This date and time is used as the time stamp that is applied to TOE generated audit records and used to track inactivity of administrative sessions.** **The time can be manually updated by a Security Administrator or automatically updated using NTP synchronization.**

If “obtain time from the underlying virtualization system” is selected, the evaluator shall examine the TSS to ensure that it identifies the VS interface the TOE uses to obtain time. If there is a delay between updates to the time on the VS and updating the time on the TOE, the TSS shall identify the maximum possible delay.

Evaluator Findings:
The ST does not select “obtain time from the underlying virtualization system” hence this assurance activity is not applicable.

Verdict:
PASS.

5.1.5.5.2 FPT_STM_EXT.1 AGD **[TD0632]**

The evaluator examines the guidance documentation to ensure it instructs the administrator how to set the time.

Evaluator Findings:
The evaluator examined the guidance documentation and ensured that it instructs the administrator how to set the time. The relevant information is found in the following section(s): **Setting Time** Upon investigation, the evaluator found that the AGD states that: **To set the system clock, the following command is needed:** **‘clock set <hh:mm:ss> [<yyyy/mm/dd>]’** Note: The time must be specified. The date is optional; if not specified, the date will be left the same. **To set the system time zone, following command is used:** **‘clock timezone <zone> [<zone word> [<zone word> [<zone word>] [<zone word>]]]’** **To display the current system time, date and time zone, following command is required:** **‘show clock’**

If the TOE supports the use of an NTP server, the guidance documentation instructs how a communication path is established between the TOE and the NTP server, and any configuration of the NTP client on the TOE to support this communication.

Evaluator Findings:
The evaluator examined the guidance documentation **section 7** and ensured that it instructs how a communication path is established between the TOE and the NTP server, and any configuration of the NTP client on the TOE to support this communication. The relevant information is found in the following section(s): **Setting Time**

Upon investigation, the evaluator found the information in AGD and paraphrasing it as:
The instructions to configure NTP are as follows:

- To enable or disable NTP overall:

  'ntp enable'

  'ntp disable'

- To add the NTP server and its version:

  'ntp server <IPv4 or IPv6 address> [version <number>]'

- To enable authentication:

  'ntp authentication enable'

- To add authentication keys:

  'ntp authentication key <key number> hash sha1 <sha1 value>'

  Note: The TOE supports authentication using SHA1 as the message digest algorithm.

- To display current NTP settings following commands are required:

  'show ntp'

  'show ntp configured'

Note: If 'configured' is specified, the configured NTP settings will be shown. If not specified, the current runtime state of NTP is given.

If the TOE supports obtaining time from the underlying VS, the evaluator shall verify the guidance documentation specifies any configuration steps necessary. If no configuration is necessary, no statement is necessary in the guidance documentation. If there is a delay between updates to the time on the VS and updating the time on the TOE, the evaluator shall ensure the guidance documentation informs the administrator of the maximum possible delay.

Evaluator Findings:
The selection "obtain time from the underlying virtualization system" is not selected in ST hence this assurance activity is not applicable.

Verdict:
PASS.

5.1.6 TOE ACCESS (FTA)

5.1.6.1 FTA_SSL_EXT.1 TSF-INITIATED SESSION LOCKING

5.1.6.1.1 FTA_SSL_EXT.1 TSS

The evaluator shall examine the TSS to determine that it details whether local administrative session locking or termination is supported and the related inactivity time period settings.

Evaluator Findings:
The evaluator examined the TSS to determine that it details whether local administrative session locking or termination is supported and the related inactivity time period settings. The relevant information is found in the following section(s): TOE Summary Specification **FTA_SSL_EXT.1**. Upon investigation, the evaluator found that the TSS states that: **A Security Administrator can configure maximum inactivity time for administrative sessions through the TOE GUI[15] and CLI interfaces.** **The configuration of inactivity periods can be configured to be anywhere from 0.25-35791 minutes and are applied on a per user interface basis.** **The configured inactivity period will be applied to both local and remote sessions in the same manner. When the interface has been idle for more than the configured period of time, the session will be terminated and will require authentication to establish a new session.**

Verdict:
PASS.

5.1.6.1.2 FTA_SSL_EXT.1 AGD

The evaluator shall confirm that the guidance documentation states whether local administrative session locking or termination is supported and instructions for configuring the inactivity time period.

Evaluator Findings:
The evaluator confirmed that the guidance documentation states whether local administrative session locking or termination is supported and instructions for configuring the inactivity time period. The relevant information is found in the following section(s): **Automatic Logout due to inactivity** Upon investigation, the evaluator found that the AGD states that: **To configure maximum inactivity times for administrative sessions (after which time the user is automatically logged out and the session is terminated (applicable for both locally connected and remote sessions):** • **For Web UI – 'webui auto-logout <minutes>'** • **For CLI – 'cli session auto-logout <minutes>'** **Note: Setting the CLI session idle timeout will simultaneously affect both the remote CLI and the local CLI interfaces.**

Verdict:
PASS.

[15] VX series models don't support Web UI Fetaure

5.1.6.2 FTA_SSL.3 TSF-INITIATED TERMINATION

5.1.6.2.1 FTA_SSL.3 TSS

The evaluator shall examine the TSS to determine that it details the administrative remote session termination and the related inactivity time period.

Evaluator Findings:
The evaluator examined the TSS and determined that it details the administrative remote session termination and the related inactivity time period. The relevant information is found in the following section(s): TOE Summary Specification **FTA_SSL.3**. Upon investigation, the evaluator found that the TSS states that: **A Security Administrator can configure maximum inactivity time for administrative sessions through the TOE GUI[16] and CLI interfaces.** **The configuration of inactivity periods can be configured to be anywhere from 0.25-35791 minutes and are applied on a per user interface basis.** **The configured inactivity period will be applied to both local and remote sessions in the same manner. When the interface has been idle for more than the configured period of time, the session will be terminated and will require authentication to establish a new session.**

Verdict:
PASS.

5.1.6.2.2 FTA_SSL.3 AGD

The evaluator shall confirm that the guidance documentation includes instructions for configuring the inactivity time period for remote administrative session termination.

Evaluator Findings:
The evaluator confirmed that the guidance documentation includes instructions for configuring the inactivity time period for remote administrative session termination. The relevant information is found in the following section(s): **Automatic Logout due to inactivity** Upon investigation, the evaluator found that the AGD states that: **To configure maximum inactivity times for administrative sessions (after which time the user is automatically logged out and the session is terminated (applicable for both locally connected and remote sessions):** • **For Web UI – 'webui auto-logout <minutes>'** • **For CLI – 'cli session auto-logout <minutes>'**

[16] VX series models don't support Web UI Fetaure

Note: Setting the CLI session idle timeout will simultaneously affect both the remote CLI and the local CLI interfaces.

Verdict:
PASS.

5.1.6.3 FTA_SSL.4 USER-INITIATED TERMINATION

5.1.6.3.1 FTA_SSL.4 TSS

The evaluator shall examine the TSS to determine that it details how the local and remote administrative sessions are terminated.

Evaluator Findings:
The evaluator examined the TSS and determined that it details how the remote administrative session (and if applicable the local administrative session) are terminated. The relevant information is found in the following section(s): TOE Summary Specification **FTA_SSL.4**. Upon investigation, the evaluator found that the TSS states that: **A Security Administrator is able to exit out of both local and remote administrative sessions using the exit command from CLI and using 'LOGOUT' option from GUI**[17]**.**

Verdict:
PASS.

5.1.6.3.2 FTA_SSL.4 AGD

The evaluator shall confirm that the guidance documentation states how to terminate a local or remote interactive session.

Evaluator Findings:
The evaluator confirmed that the guidance documentation states how to terminate a remote interactive session (and if applicable the local administrative session). The relevant information is found in the following section(s): **Logging Out** Upon investigation, the evaluator found that the AGD states that: **Following command is required to terminate an interactive session from command line:** **'hostname > exit'** **And from the Web UI, select the "Log Out" Option from the administrative interface to terminate a session.**

[17] VX series models don't support Web UI Fetaure

Verdict:
PASS.

5.1.6.4 FTA_TAB.1 DEFAULT TOE ACCESS BANNERS

5.1.6.4.1 FTA_TAB.1 TSS

The evaluator shall check the TSS to ensure that it details each administrative method of access (local and remote) available to the Security Administrator (e.g., serial port, SSH, HTTPS).

Evaluator Findings:
The evaluator checked the TSS and ensured that it details each administrative method of access (local and remote) available to the Security Administrator (e.g., serial port, SSH, HTTPS). The relevant information is found in the following section(s): TOE Summary Specification **FTA_TAB.1**. Upon investigation, the evaluator found that the TSS states that: **Security Administrators can define a custom login banner that will be displayed at the following available interfaces:** • **Local CLI** • **Remote CLI** • **Remote GUI**[18]

The evaluator shall check the TSS to ensure that all administrative methods of access available to the Security Administrator are listed and that the TSS states that the TOE is displaying an advisory notice and a consent warning message for each administrative method of access. The advisory notice and the consent warning message might be different for different administrative methods of access and might be configured during initial configuration (e.g. via configuration file).

Evaluator Findings:
The evaluator checked the TSS and ensured that all administrative methods of access available to the Security Administrator are listed and that the TSS states that the TOE is displaying an advisory notice and a consent warning message for each administrative method of access. The advisory notice and the consent warning message might be different for different administrative methods of access and might be configured during initial configuration (e.g. via configuration file). The relevant information is found in the following section(s): TOE Summary Specification **FTA_TAB.1**.

[18] VX series models don't support Web UI Fetaure

Upon investigation, the evaluator found that the TSS states that: **Security Administrators can define a custom login banner that will be displayed at the following available interfaces:**

- **Local CLI**
- **Remote CLI**
- **Remote GUI**

This banner will be displayed prior to allowing Security Administrator access through those interfaces.

The advisory notice and the consent warning message can be configured differently for remote and local access interface.

Verdict:
PASS.

5.1.6.4.2 FTA_TAB.1 AGD

The evaluator shall check the guidance documentation to ensure that it describes how to configure the banner message.

Evaluator Findings:
The evaluator examined the guidance documentation and ensured that it describes how to configure the banner message. The relevant information is found in the following section(s): **Login Banners** Upon investigation, the evaluator found that the AGD states that: **To configure local banner, requires following command:** **'banner login-local "<text>"'** **To configure Remote banner, requires following command:** **'banner login-remote "<text>"'**

Verdict:
PASS.

5.1.7 TRUSTED PATH (FTP)

5.1.7.1 FTP_ITC.1 INTER-TSF TRUSTED CHANNEL

5.1.7.1.1 FTP_ITC.1 TSS

The evaluator shall examine the TSS to determine that, for all communications with authorized IT entities identified in the requirement, each secure communication mechanism is identified in terms of the allowed protocols for that IT entity, whether the TOE acts as a server or a client, and the method of assured identification of the non-TSF endpoint.

Evaluator Findings:
The evaluator examined the TSS and determined that, for all communications with authorized IT entities identified in the requirement, each secure communication mechanism is identified in terms of the allowed protocols for that IT entity, whether the TOE acts as a server or a client, and the method of assured identification of the non-TSF endpoint. The relevant information is found in the following section(s): TOE Summary Specification **FTP_ITC.1.** Upon investigation, the evaluator found that the TSS states **that The TOE supports communication with following authorized IT entity:** • **Audit Server** **This connection is secured through a TLS connection, with the TOE acting as a TLS client. The encryption uses AES to protect the data from disclosure, and HMACs to ensure data integrity by verifying that it has not been modified.** **TLS provides assured identification of the non-TSF endpoint by validating X.509 certificates. The TOE retains a trusted store of certificate authorities which it uses to verify digital signatures on those non-TSF certificates. The TOE is responsible for initiating the trusted channel with the external trusted IT entities.**

The evaluator shall also confirm that all secure communication mechanisms are described in sufficient detail to allow the evaluator to match them to the cryptographic protocol Security Functional Requirements listed in the ST.

Evaluator Findings:
The evaluator also confirmed that all secure communication mechanisms are described in sufficient detail to allow the evaluator to match them to the cryptographic protocol Security Functional Requirements listed in the ST. The relevant information is found in the following section(s): TOE Summary Specification **FTP_ITC.1.** Upon investigation, the evaluator found that: **The TOE supports communication with following authorized IT entity:** • **Audit Server** **The connection between the TOE and the audit server is protected via a TLS connection (the TOE acts as a TLS client). This protects the data from disclosure by encryption using AES and by HMACs that verify that data has not been modified. TLS provides assured identification of the non-TSF endpoint by validating X.509 certificates.**

Verdict:
PASS.

5.1.7.1.2 FTP_ITC.1 AGD

The evaluator shall confirm that the guidance documentation contains instructions for establishing the allowed protocols with each authorized IT entity, and that it contains recovery instructions should a connection be unintentionally broken.

Evaluator Findings:
The evaluator confirmed that the AGD contains instructions for establishing the allowed protocols with each authorized IT entity, and that it contains recovery instructions should a connection be unintentionally broken. The relevant information is found in the following section(s): **TLS** Upon investigation, the evaluator found that the AGD states that: **Once the disruption has been corrected, the syslog client on the TOE will automatically attempt to re-negotiate the TLS channel upon the next retry. The syslog TLS client will transmit the Supported Elliptic Curves extension in the Client Hello message by default with support for the following NIST curves: secp256r1, secp384r1, and secp521r1. The non-TOE server can choose to negotiate the elliptic curve from this set for any of the mutually negotiable elliptic curve ciphersuites no additional configuration is required.**

Verdict:
PASS.

5.1.7.2 FTP_TRP.1/ADMIN TRUSTED PATH

5.1.7.2.1 FTP_TRP.1/ADMIN TSS

The evaluator shall examine the TSS to determine that the methods of remote TOE administration are indicated, along with how those communications are protected.

Evaluator Findings:
The evaluator examined the TSS and determined that the methods of remote TOE administration are indicated, along with how those communications are protected. The relevant information is found in the following section(s): TOE Summary Specification **FTP_TRP.1/Admin.** Upon investigation, the evaluator found that the TSS states that: **All remote administrative communications take place over a secure encrypted session.** **Remote CLI connections take place over an SSHv2 tunnel. The SSHv2 session is encrypted using AES encryption to protect confidentiality and uses HMACs to protect integrity of traffic.** **Remote GUI[19] connections take place over a HTTPS/TLS connection. The TLS session is encrypted using AES encryption and uses HMACs to protect integrity.**

[19] VX series models don't support Web UI feature.

The remote administrators can initiate both SSHv2 and HTTPS/TLS communications with the TOE.

The evaluator shall also confirm that all protocols listed in the TSS in support of TOE administration are consistent with those specified in the requirement, and are included in the requirements in the ST.

Evaluator Findings:
The evaluator also confirmed that all protocols listed in the TSS in support of TOE administration are consistent with those specified in the requirement, and are included in the requirements in the ST. The relevant information is found in the following section(s): TOE Summary Specification **FTP_TRP.1/Admin.** Upon investigation, the evaluator found that the TSS states that: **Remote CLI connections take place over an SSHv2 tunnel. The SSHv2 session is encrypted using AES encryption to protect confidentiality and uses HMACs to protect integrity of traffic.** **Remote GUI[20] connections take place over a HTTPS/TLS connection. The TLS session is encrypted using AES encryption and uses HMACs to protect integrity.**

Verdict:
PASS.

5.1.7.2.2 FTP_TRP.1/ADMIN AGD

The evaluator shall confirm that the guidance documentation contains instructions for establishing the remote administrative sessions for each supported method.

Evaluator Findings:
The evaluator confirmed that the guidance documentation contains instructions for establishing the remote administrative sessions for each supported method. The relevant information is found in the following section(s): **Enabling CC-NDcPP Compliance Mode, Remote SSH Administration** and **TLS** Upon investigation, the evaluator found that the AGD describes **all of the configuration required to establish remote TLS, HTTPS and SSH connections to the TOE.**

Verdict:
PASS.

5.2 SELECTION-BASED REQUIREMENTS

[20] VX series models don't support Web UI Fetaure

5.2.1 CRYPTOGRAPHIC SUPPORT (FCS)

5.2.1.1 FCS_HTTPS_EXT.1 HTTPS PROTOCOL[21]

5.2.1.1.1 FCS_HTTPS_EXT.1 TSS

The evaluator shall examine the TSS and determine that enough detail is provided to explain how the implementation complies with RFC 2818.

Evaluator Findings:
The evaluator examined the TSS and determine that enough detail is provided to explain how the implementation complies with RFC 2818. The relevant information is found in the following section(s): TOE Summary Specification FCS_HTTPS_EXT.1. Upon investigation, the evaluator found that the TSS states that: **The TOE does not use HTTPS in a client capacity. The TOE's HTTPS protocol complies with RFC 2818.** **RFC 2818 is HTTP over TLS. The TOE web GUI operates on an explicit TCP port designed to natively speak TLS. The web server attempts to send closure Alerts prior to closing a connection in accordance with section 2.2.2 of RFC 2818.**

Verdict:
PASS.

5.2.1.1.2 FCS_HTTPS_EXT.1 AGD

The evaluator shall examine the guidance documentation to verify it instructs the Administrator how to configure TOE for use as an HTTPS client or HTTPS server.

Evaluator Findings:
The evaluator examined the guidance documentation to verify it instructs the Administrator how to configure TOE for use as an HTTPS client or HTTPS server. The relevant information is found in the following section(s): **Configuring X.509 Certificate Authentication for the Web UI, Addition of Certificates to Trust Store** and **Reverify the web server certificate** Upon investigation, the evaluator found that the AGD describes that **the two components required to configure the HTTPS interface for the TOE, generating a certificate and installing the certificate.** **The TOE does not use HTTPS in a client capacity.**

[21]VX series models doesn't support Web UI Feature and hence this selection-based SFR is not applicable to the VX Series Models

Verdict:
PASS.

5.2.1.2 FCS_NTP_EXT.1 NTP PROTOCOL

5.2.1.2.1 FCS_NTP_EXT.1.1 TSS

The evaluator shall examine the TSS to ensure it identifies the version of NTP supported, how it is implemented and what approach the TOE uses to ensure the timestamp it receives from an NTP timeserver (or NTP peer) is from an authenticated source and the integrity of the time has been maintained.

Evaluator Findings:
The evaluator examined the TSS to ensure it identifies the version of NTP supported, how it is implemented and what approach the TOE uses to ensure the timestamp it receives from an NTP timeserver (or NTP peer) is from an authenticated source and the integrity of the time has been maintained. The relevant information is found in the following section(s): TOE Summary Specification **FCS_NTP_EXT.1.** Upon investigation, the evaluator found that the TSS states that: **The TOE supports time updates using NTPv3 and NTPv4.** **The TOE authenticates updates using an administrator configured symmetric key and SHA1.** **The TOE rejects broadcast and multicast time updates.** **With the help of configured symmetric key and SHA1 message digest algorithm ensures the timestamp it receives from an NTP timeserver is from an authenticated source and the integrity of the time has been maintained.**

The TOE must support at least one of the methods or may use multiple methods, as specified in the SFR element 1.2. the evaluator shall ensure that each method selected in the ST is described in the TSS, including the version of NTP supported in element 1.1, the message digest algorithms used to verify the authenticity of the timestamp and/or the protocols used to ensure integrity of the timestamp.

Evaluator Findings:
The TOE must support at least one of the methods or may use multiple methods, as specified in the SFR element 1.2. The evaluator shall ensure that each method selected in the ST is described in the TSS, including the version of NTP supported in element 1.1, the message digest algorithms used to verify the authenticity of the timestamp and/or the protocols used to ensure integrity of the timestamp. The relevant information is found in the following section(s): TOE Summary Specification **FCS_NTP_EXT.1.**

Upon investigation, the evaluator found that the TSS states that: **The TOE supports time updates using NTPv3 and NTPv4.**

TOE authentications updates using an administrator configured symmetric key and SHA1.**With the help of configured symmetric key and SHA1 message digest algorithm ensures the timestamp it receives from an NTP timeserver is from an authenticated source and the integrity of the time has been maintained.**

Verdict:
PASS.

5.2.1.2.2 FCS_NTP_EXT.1.1 AGD

The evaluator shall examine the guidance documentation to ensure it provides the Security Administrator instructions as how to configure the version of NTP supported, how to configure multiple NTP servers for the TOE's time source and how to configure the TOE to use the method(s) that are selected in the ST.

Evaluator Findings:

The evaluator examined the AGD to ensure it provides the Security Administrator instructions as how to configure the version of NTP supported, how to configure multiple NTP servers for the TOE's time source and how to configure the TOE to use the method(s) that are selected in the ST.

The relevant information is found in the following section(s): **Setting Time**

Upon investigation, the evaluator found that the AGD states that:

First to add an NTP server, following command is required:

'ntp server <IPv4 or IPv6 address> [version <number>]'

Note: Allowable version numbers are 3 and 4. If no version number is specified when adding a server, the default is 4.

Following command is required to enable NTP authentication prior to set algorithm:

'ntp authentication enable'

And the TOE supports authentication using SHA1 as the message digest algorithm,
following command is required to configure it:

'ntp authentication key <key number> hash sha1 <sha1 value>'

Additionally**,** AGD confirms that the TOE does not place a limit on the number of NTP time sources that can be configured, and **the above command may be used to configure multiple NTP servers for the TOE's time source.**

Verdict:
PASS.

5.2.1.2.3 FCS_NTP_EXT.1.2 AGD

For each of the secondary selections made in the ST, the evaluator shall examine the guidance document to ensure it instructs the Security Administrator how to configure the TOE to use the algorithms that support the authenticity of the timestamp and/or how to configure the TOE to use the protocols that ensure the integrity of the timestamp.

Assurance Activity Note:
Each primary selection in the SFR contains selections that specify a cryptographic algorithm or cryptographic protocol. For each of these secondary selections made in the ST, the evaluator shall examine the guidance documentation to ensure that the documentation instructs the Security Administrator how to configure the TOE to use the chosen option(s).

Evaluator Findings:
The evaluator examined the AGD and ensured that, for each of the secondary selections made in the ST, it instructs the Security Administrator how to configure the TOE to use the algorithms that support the authenticity of the timestamp and/or how to configure the TOE to use the protocols that ensure the integrity of the timestamp. The relevant information is found in the following section(s): **Setting Time** Upon investigation, the evaluator found that the AGD states that: **To configure the TOE to use the algorithms that support the authenticity of the timestamp as to enable authentication following command is required prior to configuration.** **'ntp authentication enable'** Key number should be configured here before using in **'ntp server'** command. **The TOE supports authentication using SHA1 as the message digest algorithm, following command is required to configure it:** **'ntp authentication key <key number> hash sha1 <sha1 value>'** Also, evaluator found that the AGD describes that TOE ensure the integrity of the timestamp as it states: **With the help of a configured symmetric key and SHA1 message digest algorithm ensures the timestamp it receives from an NTP timeserver is from an authenticated source and the integrity of the time has been maintained.**

Verdict:
PASS.

5.2.1.2.4 FCS_NTP_EXT.1.3 AGD

The evaluator shall examine the guidance documentation to ensure it provides the Security Administrator instructions as how to configure the TOE to not accept broadcast and multicast NTP packets that would result in the timestamp being updated.

Evaluator Findings:
The evaluator examined the AGD to ensure it provides the Security Administrator instructions as how to configure the TOE to not accept broadcast and multicast NTP packets that would result in the timestamp being updated. The relevant information is found in the following section(s): **Setting Time** Upon investigation, the evaluator found that the AGD states that: **NTP implementation does not accept broadcast or multicast NTP packets. No configuration is required**.

Verdict:
PASS.

5.2.1.3 FCS_SSHS_EXT.1. SSH SERVER

5.2.1.3.1 FCS_SSHS_EXT.1.2 TSS **[TD0631]**

The evaluator shall check to ensure that the TSS contains a list of supported public key algorithms that are accepted for client authentication and that this list is consistent with signature verification algorithms selected in FCS_COP.1/SigGen (e.g., accepting EC keys requires corresponding Elliptic Curve Digital Signature algorithm claims).

Evaluator Findings:
The evaluator checked and ensured that the TSS contains a list of supported public key algorithms that are accepted for client authentication and that this list is consistent with signature verification algorithms selected in FCS_COP.1/SigGen (e.g., accepting EC keys requires corresponding Elliptic Curve Digital Signature algorithm claims). The relevant information is found in the following section(s): TOE Summary Specification **FCS_SSHS_EXT.1.** Upon investigation, the evaluator found that the TSS states that: **The presented public key algorithm is consistent with the signature verification algorithms selected in FCS_COP.1/SigGen.** **The TOE supports the following cryptographic algorithms:** • **ssh-rsa (RSA with SHA-1), rsa-sha2-512, rsa-sha2-256**

The evaluator shall confirm that the TSS includes the description of how the TOE establishes a user identity when an SSH client presents a public key or X.509v3 certificate. For example, the TOE could verify that the SSH client's presented public key matches one that is stored within the SSH server's authorized_keys file.

Evaluator Findings:
The evaluator confirmed that the TSS includes the description of how the TOE establishes a user identity when an SSH client presents a public key or X.509v3 certificate. For example, the TOE could verify that the SSH client's presented public key matches one that is stored within the SSH server's authorized_keys file. The relevant information is found in the following section(s): TOE Summary Specification **FCS_SSHS_EXT.1.** Upon investigation, the evaluator found that the TSS states that: **The SSH server is capable of using both RSA public keys and passwords for client authentication to the remote server.** **The TOE uses username presented by the client as the user's identity. The TOE then authorizes the connection if the presented public key matches an authorized public key for the claimed identity. This is verified by confirming that the presented private key corresponds to the public key associated with the user in the 'authorized_keys' file on the TOE filesystem.**

If password-based authentication method has been selected in the FCS_SSHS_EXT.1.2, then the evaluator shall confirm its role in the authentication process is described in the TSS.

Evaluator Findings:
If password-based authentication method has been selected in the FCS_SSHS_EXT.1.2, then the evaluator confirmed its role in the authentication process is described in the TSS. The relevant information is found in the following section(s): TOE Summary Specification **FCS_SSHS_EXT.1.** Upon investigation, the evaluator found that the TSS states that**: The TOE is an SSH server, enabling administrators to remotely manage the TOE using the CLI.** **The SSH server is capable of using both RSA public keys and passwords for client authentication to the remote server.** **The password-based authentication acts as a fallback option in case the public key authentication fails.**

Verdict:
PASS.

5.2.1.3.2 FCS_SSHS_EXT.1.3 TSS

The evaluator shall check that the TSS describes how "large packets" in terms of RFC 4253 are detected and handled.

Evaluator Findings:
The evaluator checked that the TSS describes how "large packets" in terms of RFC 4253 are detected and handled.

The relevant information is found in the following section(s): TOE Summary Specification **FCS_SSHS_EXT.1.**

Upon investigation, the evaluator found that the TSS states that: **Large SSH packets are defined as those greater than 256K bytes. This is accomplished by buffering all data for a particular SSH packet transmission until the buffer limit is reached and then dropping the packet if this limit is exceeded which is inline with the RFC 4253.**

Verdict:
PASS.

5.2.1.3.3 FCS_SSHS_EXT.1.4 TSS

The evaluator shall check the description of the implementation of this protocol in the TSS to ensure that optional characteristics are specified, and the encryption algorithms supported are specified as well.

Evaluator Findings:

The evaluator checked the description of the implementation of this protocol in the TSS and ensured that optional characteristics are specified, and the encryption algorithms supported are specified as well.

The relevant information is found in the following section(s): TOE Summary Specification **FCS_SSHS_EXT.1.**

Upon investigation, the evaluator found that the TSS states that: **The TOE supports the following cryptographic algorithms:**

- **aes128-ctr, aes256-ctr, aes128-gcm@openssh.com, and aes256-gcm@openssh.com;**

The evaluator shall check the TSS to ensure that the encryption algorithms specified are identical to those listed for this component.

Evaluator Findings:

The evaluator checked the TSS and ensured that the encryption algorithms specified are identical to those listed for this component.

The relevant information is found in the following section(s): TOE Summary Specification **FCS_SSHS_EXT.1.**

Upon investigation, the evaluator found that the algorithms specified in the TSS of the ST document are identical to those listed for this component.

Verdict:
PASS.

5.2.1.3.4 FCS_SSHS_EXT.1.5 TSS **[TD0631]**

The evaluator shall check the description of the implementation of this protocol in the TSS to ensure that the SSH server's host public key algorithms supported are specified and that they are identical to those listed for this component.

Evaluator Findings:
The evaluator checked the description of the implementation of this protocol in the TSS and ensured that the SSH server's host public key algorithms supported are specified and that they are identical to those listed for this component. The relevant information is found in the following section(s): TOE Summary Specification **FCS_SSHS_EXT.1.** Upon investigation, the evaluator found that the TSS states that: **The TOE supports the following cryptographic algorithms:** • **ssh-rsa (RSA with SHA-1), rsa-sha2-512, rsa-sha2-256**

Verdict:
PASS.

5.2.1.3.5 FCS_SSHS_EXT.1.6 TSS

The evaluator shall check the TSS to ensure that it lists the supported data integrity algorithms, and that the list corresponds to the list in this component.

Evaluator Findings:
The evaluator checked the TSS and ensured that it lists the supported data integrity algorithms, and that the list corresponds to the list in this component. The relevant information is found in the following section(s): TOE Summary Specification **FCS_SSHS_EXT.1.** Upon investigation, the evaluator found that the TSS states that: **The TOE supports the following cryptographic algorithms:** • **hmac-sha2-256, hmac-sha2-512, implicit**

Verdict:
PASS.

5.2.1.3.6 FCS_SSHS_EXT.1.7 TSS

The evaluator shall check the TSS to ensure that it lists the supported key exchange algorithms, and that the list corresponds to the list in this component.

Evaluator Findings:
The evaluator checked the TSS and ensured that it lists the supported key exchange algorithms, and that the list corresponds to the list in this component. The relevant information is found in the following section(s): TOE Summary Specification **FCS_SSHS_EXT.1.** Upon investigation, the evaluator found that the TSS states that: **The TOE supports the following cryptographic algorithms:** • **diffie-hellman-group14-sha1, diffie-hellman-group14-sha256, diffie-hellman-group16-sha512, diffie-hellman-group18-sha512.**

Verdict:
PASS.

5.2.1.3.7 FCS_SSHS_EXT.1.8 TSS

The evaluator shall check that the TSS specifies the following:

a. Both thresholds are checked by the TOE.
b. Rekeying is performed upon reaching the threshold that is hit first.

Evaluator Findings:
The evaluator checked that the TSS specifies the following: a. Both thresholds are checked by the TOE. b. Rekeying is performed upon reaching the threshold that is hit first. The relevant information is found in the following section(s): TOE Summary Specification **FCS_SSHS_EXT.1.** Upon investigation, the evaluator found that the TSS states that: **TOE SSH server is capable of rekeying. The TOE implements two thresholds:** • **When 1 GB of data is transferred between using an encryption key; and** • **When 1 hour has elapsed.** **The TOE continuously checks both conditions. When either of the conditions are met, the TOE will initiate a rekey. All session keys are rekeyed at the same time (e.g. confidentiality and integrity keys).**

Verdict:
PASS.

5.2.1.3.8 FCS_SSHS_EXT.1.4 AGD

The evaluator shall also check the guidance documentation to ensure that it contains instructions on configuring the TOE so that SSH conforms to the description in the TSS (for instance, the set of algorithms advertised by the TOE may have to be restricted to meet the requirements).

Evaluator Findings:
The evaluator also checked the AGD and ensured that it contains instructions on configuring the TOE so that SSH conforms to the description in the TSS (for instance, the set of algorithms advertised by the TOE may have to be restricted to meet the requirements). The relevant information is found in the following section(s): **Details of CC Mode** Upon investigation, the evaluator found that the AGD states that: **Once NDcPP compliance is enabled below settings will be applied by default without any additional configuration changes.** • **Appliance provides AES encryption/decryption in CBC, CTR an GCM mode with 128-bit and 256-bit keys.**

Verdict:
PASS.

5.2.1.3.9 FCS_SSHS_EXT.1.5 AGD

The evaluator shall also check the guidance documentation to ensure that it contains instructions on configuring the TOE so that SSH conforms to the description in the TSS (for instance, the set of algorithms advertised by the TOE may have to be restricted to meet the requirements).

Evaluator Findings:
The evaluator also checked the AGD and ensured that it contains instructions on configuring the TOE so that SSH conforms to the description in the TSS (for instance, the set of algorithms advertised by the TOE may have to be restricted to meet the requirements). The relevant information is found in the following section(s): **Details of CC Mode** Upon investigation, the evaluator found that the AGD states that: **Once NDcPP compliance is enabled below settings will be applied by default without any additional configuration changes.** • **The Appliance provides the following cryptographic algorithms for SSH protocol.**   ○ **ssh-rsa (RSA with SHA-1), rsa-sha2-512, rsa-sha2-256**

Verdict:
PASS.

5.2.1.3.10 FCS_SSHS_EXT.1.6 AGD

The evaluator shall also check the guidance documentation to ensure that it contains instructions to the Security Administrator on how to ensure that only the allowed data integrity algorithms are used in SSH connections with the TOE (specifically, that the “none” MAC algorithm is not allowed).

Evaluator Findings:
The evaluator also checked the AGD and ensured that and ensured that it contains instructions to the Security Administrator on how to ensure that only the allowed data integrity algorithms are used in SSH connections with the TOE (specifically, that the “none” MAC algorithm is not allowed). The relevant information is found in the following section(s): **Details of CC Mode** Upon investigation, the evaluator found that the AGD states that: **Once NDcPP compliance is enabled below settings will be applied by default without any additional configuration changes.** • **Appliance implements HMAC message authentication. HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 are supported with cryptographic key sizes of 160, 256, 384, and 512 bits and message digest sizes of 160, 256, 384, and 512 bits.**

Verdict:
PASS.

5.2.1.3.11 FCS_SSHS_EXT.1.7 AGD

The evaluator shall also check the guidance documentation to ensure that it contains instructions to the Security Administrator on how to ensure that only the allowed key exchange algorithms are used in SSH connections with the TOE.

Evaluator Findings:
The evaluator also checked the AGD and ensured that it contains instructions to the Security Administrator on how to ensure that only the allowed key exchange algorithms are used in SSH connections with the TOE. The relevant information is found in the following section(s): **Details of CC Mode** Upon investigation, the evaluator found that the AGD confirms that: **Once NDcPP compliance is enabled below settings will be applied by default without any additional configuration changes.** • **Appliance provides key generation for DHG14 (2048 bits), DH16 (4096 bits), and DH18 (8192 bits) in DH key exchanges used in SSH.**

Verdict:
PASS.

5.2.1.3.12 FCS_SSHS_EXT.1.8 AGD

If one or more thresholds that are checked by the TOE to fulfil the SFR are configurable, then the evaluator shall check that the guidance documentation describes how to configure those thresholds. Either the allowed values are specified in the guidance documentation and must not exceed the limits specified in the SFR (one hour of session time, one gigabyte of transmitted traffic) or the TOE must not accept values beyond the limits specified in the SFR.

Evaluator Findings:
If one or more thresholds that are checked by the TOE to fulfil the SFR are configurable, then the evaluator checked that the AGD describes how to configure those thresholds. Either the allowed values are specified in the AGD and must not exceed the limits specified in the SFR (one hour of session time, one gigabyte of transmitted traffic) or the TOE must not accept values beyond the limits specified in the SFR. The relevant information is found in the following section(s): **Remote SSH Administration** Upon investigation, the evaluator found that the AGD states that: To enable rekey following command is required: **'ssh server rekey enable'** **To configure data threshold limit, following command is required:** **'ssh server rekey data-limit <data limit in MB>'** Note: Data limit is one gigabyte. **To configure time limit, following command is required:** **'ssh server rekey time-limit <time limit in seconds>'** Note: Time limit is one hour. **This command enables and sets data and time limits when the server will force the session key to be renegotiated.**

The evaluator shall check that the guidance documentation describes that the TOE reacts to the first threshold reached.

Evaluator Findings:
The evaluator checked that the AGD describes that the TOE reacts to the first threshold reached. The relevant information is found in the following section(s): **Remote SSH Administration** Upon investigation, the evaluator found that the AGD states that: **The TOE continuously checks both conditions. When either of the conditions are met, the TOE will initiate a rekey. All session keys are rekeyed at the same time (e.g. confidentiality and integrity keys).**

Verdict:
PASS.

5.2.1.4 FCS_TLSC_EXT.1 EXTENDED: TLS CLIENT PROTOCOL WITHOUT MUTUAL AUTHENTICATION

5.2.1.4.1 FCS_TLSC_EXT.1.1 TSS

The evaluator shall check the description of the implementation of this protocol in the TSS to ensure that the ciphersuites supported are specified.

Evaluator Findings:
The evaluator checked the description of the implementation of this protocol in the TSS and ensured that the ciphersuites supported are specified. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSC_EXT.1.** Upon investigation, the evaluator found that the TSS states that: **The syslog channel client supports TLS protocol version 1.2 and are restricted to the following ciphersuites:** • **TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268** • **TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268** • **TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256 as defined in RFC 5246** • **TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246** • **TLS_ECDHE_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 evaluator shall check the TSS to ensure that the ciphersuites specified include those listed for this component.

Evaluator Findings:
The evaluator checked the TSS and ensured that the ciphersuites specified include those listed for this component. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSC_EXT.1.** Upon investigation, the evaluator found that the ciphersuites specified in the TSS of the ST document are identical to those listed for this component.

Verdict:
PASS.

5.2.1.4.2 FCS_TLSC_EXT.1.2 TSS

The evaluator shall ensure that the TSS describes the client's method of establishing all reference identifiers from the administrator/application- configured reference identifier, including which types of reference identifiers are supported (e.g. application-specific Subject Alternative Names) and whether IP addresses and wildcards are supported.

Evaluator Findings:
The evaluator ensured that the TSS describes the client's method of establishing all reference identifiers from the administrator/application- configured reference identifier, including which types of reference identifiers are supported (e.g. application-specific Subject Alternative Names) and whether IP addresses and wildcards are supported. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSC_EXT.1.** Upon investigation, the evaluator found that the TSS states that: **The reference identifier for the syslog server is configured by the administrator using the available administrative commands in the CLI.** **The reference identifiers must be an IPv4 address, IPv6 address, or a hostname(FQDN).** **When the reference identifier is a hostname, the TOE compares the hostname against all of the entries in the Subject Alternative Name extension. If the hostname does not match any of the entries, then the verification fails. If the certificate does not contain any entries in the SAN, the TSF will continue to compare the hostname against the Common Name (CN). If the hostname does not match the CN, then the verification fails.** **For both SAN and CN, the hostname must be an exact match or wildcard match. In the case of a wildcard match; the wildcard must be the left-most component, wildcard matches a single component, and there are at least two non-wildcard components.**

Note that where a TLS channel is being used between components of a distributed TOE for FPT_ITT.1, the requirements to have the reference identifier established by the user are relaxed and the identifier may also be established through a "Gatekeeper" discovery process. The TSS should describe the discovery process and highlight how the reference identifier is supplied to the "joining" component. Where the secure channel is being used between components of a distributed TOE for FPT_ITT.1 and the ST author selected attributes from RFC 5280, the evaluator shall ensure the TSS describes which attribute type, or combination of attributes types, are used by the client to match the presented identifier with the configured identifier. The evaluator shall ensure the TSS presents an argument how the attribute type, or combination of attribute types, uniquely identify the remote TOE component; and the evaluator shall verify the attribute type, or combination of attribute types, is sufficient to support unique identification of the maximum supported number of TOE components.

Evaluator Findings:
The TOE is not a distributed TOE; hence this assurance activity is not applicable.

If IP addresses are supported in the CN as reference identifiers, the evaluator shall ensure that the TSS describes the TOE's conversion of the text representation of the IP address in the CN to a binary representation of the IP address in network byte order.

Evaluator Findings:
If IP addresses are supported in the CN as reference identifiers, the evaluator ensured that the TSS describes the TOE's conversion of the text representation of the IP address in the CN to a binary representation of the IP address in network byte order. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSC_EXT.1**. Upon investigation, the evaluator found that the TSS states that: **When the reference identifier is an IP address, the TOE converts the IP address to a binary representation in network byte order.** **IPv4 addresses are converted directly from decimal to binary, IPv6 addresses are converted as specified in RFC 5952.** **The TOE compares the binary IP address against all of the IP Address entries in the Subject Alternative Name extension. If there is not an exact binary match, then the verification fails.**

The evaluator shall also ensure that the TSS describes whether canonical format (RFC 5952 for IPv6, RFC 3986 for IPv4) is enforced.

Evaluator Findings:
The evaluator also ensured that the TSS describes whether canonical format (RFC 5952 for IPv6, RFC 3986 for IPv4) is enforced. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSC_EXT.1**. Upon investigation, the evaluator found that the TSS states that: **IPv4 addresses are converted directly from decimal to binary as specified in RFC 3986, IPv6 addresses are converted as specified in RFC 5952.**

Verdict:
PASS.

5.2.1.4.3 FCS_TLSC_EXT.1.4 TSS

The evaluator shall verify that TSS describes the Supported Elliptic Curves/Supported Groups Extension and whether the required behaviour is performed by default or may be configured.

Evaluator Findings:
The evaluator verified that TSS describes the Supported Elliptic Curves/Supported Groups Extension and whether the required behaviour is performed by default or may be configured. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSC_EXT.1.** Upon investigation, the evaluator found that the TSS states that: **The syslog TLS client will transmit the Supported Elliptic Curves extension in the Client Hello message by default with support for the following NIST curves: secp256r1, secp384r1, and secp521r1.** **The non-TOE server can choose to negotiate the elliptic curve from this set for any of the mutually negotiable elliptic curve cipher suites and no additional configuration is required.**

Verdict:
PASS.

5.2.1.4.4 FCS_TLSC_EXT.1.1 AGD

The evaluator shall check the guidance documentation to ensure that it contains instructions on configuring the TOE so that TLS conforms to the description in the TSS.

Evaluator Findings:
The evaluator checked the AGD and ensured that it contains instructions on configuring the TOE so that TLS conforms to the description in the TSS.

The relevant information is found in the following section(s): **TLS** and **Details of CC Mode**

Upon investigation, the evaluator found that the AGD confirms that:
No configuration is required other than enabling CC-NDcPP compliance for TLS conforms to description in the TSS.

Once NDcPP compliance is enabled below settings will be applied by default without any additional configuration changes.

- **Appliance provides AES encryption/decryption in CBC, CTR an GCM mode with 128-bit and 256-bit keys.**
  - **AES is implemented in the following protocols: TLS and SSH**
- **Appliance supports signature generation and verification for RSA (2048 and 3072 bits) and ECDSA (P-256, P-384, P-521), in accordance with FIPS PUB 186-4.**
  - **RSA signature generation and verification are used for the TLS and SSH protocols**
  - **ECDSA signature verification is used in TLS**
- **It provides cryptographic hashing services for key generation using SHA-1, SHA-256, SHA-384, and SHA-512 as specified in ISO/IEC 10118-3:2004.**
  - **TLS and SSH - SHA1, SHA-256, SHA-384 and SHA-512**

Verdict:
PASS.

5.2.1.4.5 FCS_TLSC_EXT.1.2 AGD

The evaluator shall ensure that the operational guidance describes all supported identifiers, explicitly states whether the TOE supports the SAN extension or not and includes detailed instructions on how to configure the reference identifier(s) used to check the identity of peer(s). If the identifier scheme implemented by the TOE includes support for IP addresses, the evaluator shall ensure that the operational guidance provides a set of warnings and/or CA policy recommendations that would result in secure TOE use.

Evaluator Findings:
The evaluator ensured that the AGD describes all supported identifiers, explicitly states whether the TOE supports the SAN extension or not and includes detailed instructions on how to configure the reference identifier(s) used to check the identity of peer(s). The evaluator ensured that the AGD provides a set of warnings and/or CA policy recommendations that would result in secure TOE use when the identifier scheme implemented by the TOE includes support for IP addresses. The relevant information is found in the following section(s): **Using an Audit Server** and **Reference Identifiers**

Upon investigation, the evaluator found that the AGD states that:

Configuration of the reference identifiers used to check the identity of peer, following command is required to configure:

'logging <reference identifier> protocol tls port 6514'

AGD describes all supported identifiers as it states that the reference identifiers must be an IPv4 address, IPv6 address, or a hostname.

TOE supports the SAN extension. If the FQDN does not match any of the DNS Name entries in SAN, then the verification fails. If no SAN, then will compare against the CN. If does not match, then the verification fails.

Additionally, **the identifier scheme implemented by the TOE includes support for IP addresses as well. The TOE compares the binary IP address against all the IP Address entries in the SAN extension. If there is not an exact binary match, then the verification fails.**

The TLS channel is terminated if verification fails.

Note (from RFC 6125): IP addresses are not necessarily reliable identifiers for application services because of the existence of private internets [PRIVATE], host mobility, multiple interfaces on a given host, Network Address Translators (NATs) resulting in different addresses for a host from different locations on the network, the practice of grouping many hosts together behind a single IP address, etc.

Where the secure channel is being used between components of a distributed TOE for FPT_ITT.1, the SFR selects attributes from RFC 5280, and FCO_CPC_EXT.1.2 selects "no channel"; the evaluator shall verify the AGD provides instructions for establishing unique reference identifiers based on RFC5280 attributes.

Evaluator Findings:
The TOE is not a distributed TOE; hence this assurance activity is not applicable.

Verdict:
PASS.

5.2.1.4.6 FCS_TLSC_EXT.1.4 AGD

If the TSS indicates that the Supported Elliptic Curves/Supported Groups Extension must be configured to meet the requirement, the evaluator shall verify that the AGD guidance includes configuration of the Supported Elliptic Curves/Supported Groups Extension.

Evaluator Findings:
The evaluator verified that the AGD includes configuration of the Supported Elliptic Curves/Supported Groups Extension. The relevant information is found in the following section(s): **TLS** Upon investigation, the evaluator found that the AGD states that: **Syslog TLS client will transmit the Supported Elliptic Curves extension in the Client Hello message by default with support for the**

following NIST curves: secp256r1, secp384r1, and secp521r1. The non-TOE server can choose to negotiate the elliptic curve from this set for any of the mutually negotiable elliptic curve ciphersuites, no additional configuration is required.

Verdict:
PASS.

5.2.1.5 FCS_TLSS_EXT.1 EXTENDED: TLS SERVER PROTOCOL WITHOUT MUTUAL AUTHENTICATION[22]

5.2.1.5.1 FCS_TLSS_EXT.1.1 TSS

The evaluator shall check the description of the implementation of this protocol in the TSS to ensure that the ciphersuites supported are specified.

Evaluator Findings:

The evaluator checked the description of the implementation of this protocol in the TSS and ensured that the ciphersuites supported are specified.

The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSS_EXT.1.**

Upon investigation, the evaluator found that the TSS states that: **The server only supports TLS protocol version 1.2 (rejecting any other protocol version, including SSL 2.0, SSL 3.0 and TLS 1.0, TLS 1.1 and any other unknown TLS version string supplied) and is restricted to the following ciphersuites by default:**

- **TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268**
- **TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268**
- **TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256 as defined in RFC 5246**
- **TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246**
- **TLS_ECDHE_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 evaluator shall check the TSS to ensure that the ciphersuites specified are identical to those listed for this component.

[22]VX series models doesn't support Web UI Feature and hence this selection-based SFR is not applicable to the VX Series Models

Evaluator Findings:
The evaluator checked the TSS and ensured that the ciphersuites specified are identical to those listed for this component. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSS_EXT.1.** Upon investigation, the evaluator found that the ciphersuites specified in the TSS of the ST document are identical to those listed for this component.

Verdict:
PASS.

5.2.1.5.2 FCS_TLSS_EXT.1.2 TSS

The evaluator shall verify that the TSS contains a description of how the TOE technically prevents the use of old SSL and TLS versions.

Evaluator Findings:
The evaluator verified that the TSS contains a description of how the TOE technically prevents the use of old SSL and TLS versions. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSS_EXT.1.** Upon investigation, the evaluator found that the TSS states that: **The server only supports TLS protocol version 1.2 (rejecting any other protocol version, including SSL 2.0, SSL 3.0 and TLS 1.0, TLS 1.1 and any other unknown TLS version string supplied).** This protocol is implemented in all the models except the VX series models and hence this SFR is not applicable to the VX Series Models.

Verdict:
PASS.

5.2.1.5.3 FCS_TLSS_EXT.1.3 TSS **[TD0635]**

If using ECDHE and/or DHE ciphers, the evaluator shall verify that the TSS lists all EC Diffie-Hellman curves and/or Diffie-Hellman groups used in the key establishment by the TOE when acting as a TLS Server. For example, if the TOE supports TLS_DHE_RSA_WITH_AES_128_CBC_SHA cipher and Diffie-Hellman parameters with size 2048 bits, then list Diffie-Hellman Group 14.

Evaluator Findings:
The evaluator verified that the TSS lists all EC Diffie-Hellman curves and/or Diffie-Hellman groups used in the key establishment by the TOE when acting as a TLS Server.

The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSS_EXT.1.**

Upon investigation, the evaluator found that the TSS states that: **The TLS server is capable of negotiating ciphersuites that include RSA, DHE, and ECDHE key agreement schemes.**

The DHE key agreement parameters as per 'Finite field-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography"' are restricted to DHG14 (2048 bits) and are hardcoded into the server.

The ECDHE key agreement parameters are restricted to secp256r1, secp384r1, and secp521r1.

This protocol is implemented in all the models except the VX series models and hence this SFR is not applicable to the VX Series Models.

Verdict:
PASS.

5.2.1.5.4 FCS_TLSS_EXT.1.4 TSS **[TD0569]**

The evaluator shall verify that the TSS describes if session resumption based on session IDs is supported (RFC 4346 and/or RFC 5246) and/or if session resumption based on session tickets is supported (RFC 5077).

Evaluator Findings:

The evaluator verified that the TSS describes if session resumption based on session IDs is supported (RFC 4346 and/or RFC 5246) and/or if session resumption based on session tickets is supported (RFC 5077).

The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSS_EXT.1.**

Upon investigation, the evaluator found that the TSS states that**: TOE supports session resumption of the single HTTPS context using session tickets.**

Session tickets are structured as specified in Section 4 of RFC 5077 and encrypted using AES with a 128-bit key.

This protocol is implemented in all the models except the VX series models and hence this SFR is not applicable to the VX Series Models.

If session tickets are supported, the evaluator shall verify that the TSS describes that the session tickets are encrypted using symmetric algorithms consistent with FCS_COP.1/DataEncryption.

Evaluator Findings:
The evaluator verified that the TSS describes that the session tickets are encrypted using symmetric algorithms consistent with FCS_COP.1/DataEncryption. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSS_EXT.1.** Upon investigation, the evaluator found that the TSS states that: **The session tickets are encrypted using symmetric algorithm AES with a 128-bit key and are consistent with FCS_COP.1/DataEncryption.** This protocol is implemented in all the models except the VX series models and hence this SFR is not applicable to the VX Series Models.

The evaluator shall verify that the TSS identifies the key lengths and algorithms used to protect session tickets.

Evaluator Findings:
The evaluator verified that the TSS identifies the key lengths and algorithms used to protect session tickets. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSS_EXT.1.** Upon investigation, the evaluator found that the TSS states that: **The session tickets are encrypted using symmetric algorithm AES with a 128-bit key.** This protocol is implemented in all the models except the VX series models and hence this SFR is not applicable to the VX Series Models.

If session tickets are supported, the evaluator shall verify that the TSS describes that session tickets adhere to the structural format provided in Section 4 of RFC 5077 and if not, a justification shall be given of the actual session ticket format.

Evaluator Findings:
The evaluator verified that the TSS describes that session tickets adhere to the structural format provided in Section 4 of RFC 5077 and if not, a justification was given of the actual session ticket format. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSS_EXT.1.**

Upon investigation, the evaluator found that the TSS states that: **Session tickets are structured as specified in Section 4 of RFC 5077 and encrypted using AES with a 128-bit key.**

This protocol is implemented in all the models except the VX series models and hence this SFR is not applicable to the VX Series Models.

If the TOE claims a (D)TLS server capable of session resumption (as a single context, or across multiple contexts), the evaluator shall verify that the TSS describes how session resumption operates (i.e. what would trigger a full handshake, e.g. checking session status, checking Session ID, etc.). If multiple contexts are used the TSS describes how session resumption is coordinated across those contexts. In case session establishment and session resumption are always using a separate context, the TSS shall describe how the contexts interact with respect to session resumption (in particular regarding the session ID). It is acceptable for sessions established in one context to be resumable in another context.

Evaluator Findings:

The TOE claims a TLS server capable of session resumption. The evaluator verified that the TSS describes how session resumption operates.

The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSS_EXT.1.**

Upon investigation, the evaluator found that the TSS states that:

The TOE supports session resumption of the single HTTPS context using session tickets. The session tickets are encrypted using symmetric algorithm AES with a 128-bit key and are consistent with FCS_COP.1/DataEncryption. Session tickets are structured as specified in Section 4 of RFC 5077 and encrypted using AES with a 128-bit key.

Verdict:
PASS.

5.2.1.5.5 FCS_TLSS_EXT.1.1 AGD

The evaluator shall check the guidance documentation to ensure that it contains instructions on configuring the TOE so that TLS conforms to the description in the TSS (for instance, the set of ciphersuites advertised by the TOE may have to be restricted to meet the requirements).

Evaluator Findings:

The evaluator checked the AGD and ensured that it contains instructions on configuring the TOE so that TLS conforms to the description in the TSS (for instance, the set of ciphersuites advertised by the TOE may have to be restricted to meet the requirements).

The relevant information is found in the following section(s): **Details of CC Mode**

Upon investigation, the evaluator found that the AGD states that:

Once NDcPP compliance is enabled below settings will be applied by default without any additional configuration changes.

- **Appliance provides AES encryption/decryption in CBC, CTR an GCM**
- **mode with 128-bit and 256-bit keys.**
  - **AES is implemented in the following protocols: TLS and SSH**
- **Appliance supports signature generation and verification for RSA (2048 and 3072 bits) and ECDSA (P-256, P-384, P-521), in accordance with FIPS PUB 186-4.**
  - **RSA signature generation and verification are used for the TLS and SSH protocols**
  - **ECDSA signature verification is used in TLS**
- **It provides cryptographic hashing services for key generation using SHA-1, SHA-256, SHA-384, and SHA-512 as specified in ISO/IEC 10118-3:2004.**
  - **TLS and SSH - SHA1, SHA-256, SHA-384 and SHA-512**

This protocol is implemented in all the models except the VX series models and hence this SFR is not applicable to the VX Series Models.

Verdict:
PASS.

5.2.1.5.6 FCS_TLSS_EXT.1.2 AGD

The evaluator shall verify that any configuration necessary to meet the requirement must be contained in the AGD guidance.

Evaluator Findings:
The evaluator verified that any configuration necessary to meet the requirement must be contained in the AGD. The relevant information is found in the following section(s): **Details of CC Mode** Upon investigation, the evaluator found that the AGD states that: **Once NDcPP compliance is enabled below settings will be applied by default without any additional configuration changes.** • **The server supports TLS protocol version 1.2 (rejecting any other protocol version, including SSL 2.0, SSL 3.0 and TLS 1.0, TLS 1.1 and any other unknown TLS version string supplied).** This protocol is implemented in all the models except the VX series models and hence this SFR is not applicable to the VX Series Models.

Verdict:
PASS.

5.2.1.5.7 FCS_TLSS_EXT.1.3 AGD

The evaluator shall verify that any configuration necessary to meet the requirement must be contained in the AGD guidance.

Evaluator Findings:
The evaluator verified that any configuration necessary to meet the requirement must be contained in the AGD. The relevant information is found in the following section(s): **Details of CC Mode** Upon investigation, the evaluator found that the AGD states that: **Once NDcPP compliance is enabled below settings will be applied by default without any additional configuration changes.** • The server supports TLS protocol version 1.2 (rejecting any other protocol version, including SSL 2.0, SSL 3.0 and TLS 1.0, TLS 1.1 and any other unknown TLS version string supplied). This protocol is implemented in all the models except the VX series models and hence this SFR is not applicable to the VX Series Models.

Verdict:
PASS.

5.2.1.5.8 FCS_TLSS_EXT.1.4 AGD **[TD0569]**

The evaluator shall verify that any configuration necessary to meet the requirement must be contained in the AGD guidance.

Evaluator Findings:
The evaluator verified that any configuration necessary to meet the requirement must be contained in the AGD. The relevant information is found in the following section(s): **Details of CC Mode** Upon investigation, the evaluator found that the AGD states that: **Once NDcPP compliance is enabled below settings will be applied by default without any additional configuration changes.** • The server supports TLS protocol version 1.2 (rejecting any other protocol version, including SSL 2.0, SSL 3.0 and TLS 1.0, TLS 1.1 and any other unknown TLS version string supplied).

This protocol is implemented in all the models except the VX series models and hence this SFR is not applicable to the VX Series Models.

Verdict:
PASS.

5.2.2 IDENTIFICATION AND AUTHENTICATION (FIA)

5.2.2.1 FIA_X509_EXT.1/REV X.509 CERTIFICATE VALIDATION

5.2.2.1.1 FIA_X509_EXT.1/REV TSS

The evaluator shall ensure the TSS describes where the check of validity of the certificates takes place, and that the TSS identifies any of the rules for extendedKeyUsage fields (in FIA_X509_EXT.1.1) that are not supported by the TOE (i.e. where the ST is therefore claiming that they are trivially satisfied). It is expected that revocation checking is performed when a certificate is used in an authentication step and when performing trusted updates (if selected). It is not necessary to verify the revocation status of X.509 certificates during power-up self-tests (if the option for using X.509 certificates for self-testing is selected).

Evaluator Findings:

The evaluator ensured the TSS describes where the check of validity of the certificates takes place, and that the TSS identifies any of the rules for extendedKeyUsage fields (in FIA_X509_EXT.1.1) that are not supported by the TOE (i.e. where the ST is therefore claiming that they are trivially satisfied).

The relevant information is found in the following section(s): TOE Summary Specification **FIA_X509_EXT.1/Rev.**

Upon investigation, the evaluator found that the TSS states that:

The TOE performs X.509 certificate validation at the following points:TOE TLS client authentication of server X.509 certificates; When certificates are loaded into the TOE, such as when importing CAs, certificate responses and other device-level certificates (such as the web server certificate presented by the TOE TLS web GUI). In all scenarios, Certificates are checked for several validation characteristics:eIf the certificate 'notAfter' date is in the past, then this is an expired certificate which is considered invalid;eIf the certificate 'notBefore' date is in the future, then the certificate is considered invalid;eThe certificate chain must terminate with a trusted CA certificate;

- **Server certificates consumed by the TOE TLS client must have a 'serverAuthentication' extendedKeyUsage purpose**

A trusted CA certificate is defined as any certificate loaded into the TOE trust store that has, at a minimum, a basicConstraints extension with the CA flag set to TRUE.

As X.509 certificates are not used for either trusted updates or firmware integrity self-tests, the code-signing purpose is not checked for in the extendedKeyUsage, hence the requirement for Code

Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field is trivially satisfied.

Certificate revocation checking is performed on the leaf and intermediate CA certificates using OCSP responders as a part of authentication step. The OCSP signing certificate must have the OCSP signing purpose in the extendedKeyUsage extension.

The TSS shall describe when revocation checking is performed and on what certificates. If the revocation checking during authentication is handled differently depending on whether a full certificate chain or only a leaf certificate is being presented, any differences must be summarized in the TSS section and explained in the Guidance.

Evaluator Findings:

The TSS describes when revocation checking is performed and on what certificates. Any differences where revocation checking during authentication is handled differently depending on whether a full certificate chain or only a leaf certificate is being presented is summarized in the TSS section and explained in the Guidance.

The relevant information is found in the following section(s): TOE Summary Specification **FIA_X509_EXT.1/Rev.**

Upon investigation, the evaluator found that the TSS states that: **Certificate revocation checking is performed on the leaf and intermediate CA certificates using OCSP responders as a part of authentication step.**

There is no difference in handling of revocation checking during authentication irrespective of whether a full certificate chain or only a leaf certificate is being presented.

The OCSP signing certificate must have the OCSP signing purpose in the extendedKeyUsage extension.

Verdict:
PASS.

5.2.2.1.2 FIA_X509_EXT.1/REV AGD

The evaluator shall also ensure that the guidance documentation describes where the check of validity of the certificates takes place, describes any of the rules for extendedKeyUsage fields (in FIA_X509_EXT.1.1) that are not supported by the TOE (i.e. where the ST is therefore claiming that they are trivially satisfied) and describes how certificate revocation checking is performed and on which certificate.

Evaluator Findings:

The evaluator also ensured that the AGD describes where the check of validity of the certificates takes place, describes any of the rules for extendedKeyUsage fields (in FIA_X509_EXT.1.1) that are not supported by the TOE (i.e. where the ST is therefore claiming that they are trivially satisfied) and describes how certificate revocation checking is performed and on which certificate.

The relevant information is found in the following section(s): **X.509 Certificate**

Upon investigation, the evaluator found that the AGD states that:

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

- **TOE TLS client authentication of server X.509 certificates.**
- **When certificates are loaded into the TOE, such as when importing CAs, certificate responses and other device-level certificates (such as the web server certificate presented by the TOE TLS web GUI).**

And The TOE validates certificates in accordance with the following rules:

- **RFC 5280 certificate validation and certification path validation supporting a minimum path length of three certificates.**
- **The certification path must terminate with a trusted CA certificate designated as a trust anchor.**
- **The TOE validates 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 TOE validates the revocation status of the certificate using Online Certificate Status Protocol (OCSP) as specified in RFC 6960.**
- **The TOE validates the extendedKeyUsage field according to the following rules:**
    - **Server certificates presented for TLS must have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.**
    - **Client certificates presented for TLS must have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.**
    - **OCSP certificates presented for OCSP responses shall 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 does not use X.509 certificates for trusted updates, hence the requirement for Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field is trivially satisfied.

Certificate revocation checking is performed on the leaf and intermediate CA certificates using OCSP responders as part of the authentication step. There is no difference in handling of revocation checking during authentication irrespective of whether a full certificate chain or only a leaf certificate is being presented. The OCSP signing certificate must have the OCSP signing purpose in the extendedKeyUsage extension.

Verdict:
PASS.

5.2.2.2 FIA_X509_EXT.2 X.509 CERTIFICATE AUTHENTICATION

5.2.2.2.1 FIA_X509_EXT.2 TSS

The evaluator shall check the TSS to ensure that it describes how the TOE chooses which certificates to use, and any necessary instructions in the administrative guidance for configuring the operating environment so that the TOE can use the certificates.

Evaluator Findings:
The evaluator checked the TSS and ensured that it describes how the TOE chooses which certificates to use, and any necessary instructions in the AGD for configuring the operating environment so that the TOE can use the certificates. The relevant information is found in the following section(s): TOE Summary Specification **FIA_X509_EXT.2** Upon investigation, the evaluator found that the TSS states that: **The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for all TLS and HTTPS peer entities. Certificates are used to authenticate and establish a secure communication channel for the audit server.** **The TOE allows each TLS service to be configured with its certificate in the TLS profile. Once the certificate is configured for an audit server using a TLS profile, that certificate will be used for all audit server connection authentication.** **Instructions for configuring the trusted IT entities to supply appropriate X.509 certificates are captured in the guidance documents.**

The evaluator shall examine the TSS and confirmed that it describes the behaviour of the TOE when a connection cannot be established during the validity check of a certificate used in establishing a trusted channel.

Evaluator Findings:
The evaluator examined the TSS and confirmed that it describes the behaviour of the TOE when a connection cannot be established during the validity check of a certificate used in establishing a trusted channel. The relevant information is found in the following section(s): TOE Summary Specification **FIA_X509_EXT.2** Upon investigation, the evaluator found that the TSS states that: **If, during the entire trust chain verification activity, any certificate under review fails a verification check, then the entire trust chain is deemed untrusted and the TLS connection is terminated, as TLS is only trusted channel.** **As part of the verification process, OCSP is used to determine whether the certificate is revoked or not. If the OCSP responder cannot be contacted, then the TOE will choose to automatically reject the certificate in this case.**

The evaluator shall verify that any distinctions between trusted channels are described. If the requirement that the administrator is able to specify the default action, then the evaluator shall ensure that the AGD contains instructions on how this configuration action is performed.

Evaluator Findings:
The evaluator verified that any distinctions between trusted channels are described. The evaluator ensured that the guidance documentation contains instructions on how the administrator is able to specify the default action. The relevant information is found in the following section(s): TOE Summary Specification **FIA_X509_EXT.2** Upon investigation, the evaluator found that the TSS states that: **The administrator does not determine the default handling of certificates.**

Verdict:
PASS.

5.2.2.2.2 FIA_X509_EXT.2 AGD

The evaluator shall also ensure that the guidance documentation describes the configuration required in the operating environment so the TOE can use the certificates. The guidance documentation shall also include any required configuration on the TOE to use the certificates. The guidance document shall also describe the steps for the Security Administrator to follow if the connection cannot be established during the validity check of a certificate used in establishing a trusted channel.

Evaluator Findings:
The evaluator also ensured that the AGD describes the configuration required in the operating environment so the TOE can use the certificates. The AGD also includes any required configuration on the TOE to use the certificates. The AGD also describes the steps for the Security Administrator to follow if the connection cannot be established during the validity check of a certificate used in establishing a trusted channel. The relevant information is found in the following section(s): **Configuring X.509 Certificate Authentication for the Web UI**, **Addition of Certificates to Trust Store** and **Audit Server Configuration** Upon investigation, the evaluator found that the AGD states that: To add certificates using web UI, following steps are required: • **On the Web UI[23], select Settings Tab** • **Select Certificates/Keys** • **Click Add Root/Intermediate CA Certificate** • **Choose file then commit** To add certificate using CLI, following commands are required: **'crypto certificate name xxx public-cert pem xxx'**

[23] VX series models don't support Web UI Fetaure

'crypto certificate ca-list default-ca-list name xxx'

If a connection is not possible because the validity of a certificate cannot be determined, there is no override option. A valid certificate must be presented. This may include installing required certificates in the trust store.

To enable OCSP checking run the below command:

'hostname (config) # logging remote OCSP enable'

The OCSP Server, provided by the operational environment, must be loaded with the following certificates:

- **Self-certificate (system cert) signed by the issuer (CA authority)**
- **Root certificate who signed the system certificate**
- **Root certificate of the client who is trying to initiate the connection**

Verdict:
PASS.

5.2.2.3 FIA_X509_EXT.3 EXTENDED: X509 CERTIFICATE REQUESTS

5.2.2.3.1 FIA_X509_EXT.3 TSS

If the ST author selects "device-specific information", the evaluator shall verify that the TSS contains a description of the device-specific fields used in certificate requests.

Evaluator Findings:
The evaluator verified that the TSS contains a description of the device-specific fields used in certificate requests. The relevant information is found in the following section(s): TOE Summary Specification **FIA_X509_EXT.3.** Upon investigation, the evaluator found that the TSS states that**: No device-specific details are collected and added to the certificate request to be signed.**

Verdict:
PASS.

5.2.2.3.2 FIA_X509_EXT.3 AGD

The evaluator shall check to ensure that the guidance documentation contains instructions on requesting certificates from a CA, including generation of a Certificate Request. If the ST author selects "Common Name", "Organization", "Organizational Unit", or "Country", the evaluator shall ensure that this guidance includes instructions for establishing these fields before creating the Certification Request.

Evaluator Findings:
The evaluator checked and ensured that the AGD contains instructions on requesting certificates from a CA, including generation of a Certificate Request. The evaluator ensured that the AGD includes

instructions for establishing the Common name, Organization, Organizational unit and Country code fields before creating the Certification Request.

The relevant information is found in the following section(s): **Configuring X.509 Certificate Authentication for the Web UI**[24]

Upon investigation, the evaluator found that the AGD states that:

For generating and to issue CSR following command is required:

'crypto certificate signing-request generate'

The above command generates a CSR without the optional common name. To generate a CSR with a common name, the request must be made with the following option,

- **Name** – This is the common name of the device
- **Organization** – This is the associated organization
- **Org-Unit** –This is the associated organizational-Unit
- **Country-Code** –This is the associated Country

After a certificate is generated from an external server, the full path certificate must be uploaded to the TOE using the following command,

Crypto certificate name <name of the certificate> public-cert match csr <name of the CSR>

The full public certificate must then be copied to the command line.

Verdict:
PASS.

5.2.3 SECURITY MANAGEMENT (FMT)

5.2.3.1 FMT_MOF.1/FUNCTIONS MANAGEMENT OF SECURITY FUNCTIONS BEHAVIOUR

5.2.3.1.1 FMT_MOF.1/FUNCTIONS TSS

For distributed TOEs see Section 2.4.1.1.

Evaluator Findings:
The TOE is not distributed; hence, this requirement is not applicable.

For non-distributed TOEs, the evaluator shall ensure the TSS for each administrative function identified the TSS details how the Security Administrator determines or modifies the behaviour of (whichever is supported by the TOE) transmitting audit data to an external IT entity, handling of audit data, audit functionality when Local Audit Storage Space is full (whichever is supported by the TOE).

[24] VX series models don't support Web UI Fetaure

Evaluator Findings:
The evaluator examined the TSS and ensured that, for non-distributed TOEs, it details how the Security Administrator modifies the behaviour of transmitting audit data to an external IT entity, handling of audit data, audit functionality when Local Audit Storage Space is full. The relevant information is found in the following section(s): **FMT_MOF.1/Functions** Upon investigation, the evaluator found that the TSS states that: **TOE restricts the ability to modify the behavior of transmission of audit data to an external IT entity (Audit Server (FQDN or IP address), OCSP responder, TLS ciphersuites), handling of audit data (number of logs to retain) to Security Administrators.**

Verdict:
PASS.

5.2.3.1.2 FMT_MOF.1/FUNCTIONS AGD

For distributed TOEs see Section 2.4.1.2.

Evaluator Findings:
The TOE is not distributed; hence, this requirement is not applicable.

For non-distributed TOEs, the evaluator shall also ensure the Guidance Documentation describes how the Security Administrator determines or modifies the behaviour of (whichever is supported by the TOE) transmitting audit data to an external IT entity, handling of audit data, audit functionality when Local Audit Storage Space is full (whichever is supported by the TOE) are performed to include required configuration settings.

Evaluator Findings:
The evaluator examined the AGD and ensured that, for non-distributed TOEs, it describes how the Security Administrator determines or modifies the behaviour of transmitting audit data to an external IT entity, handling of audit data, audit functionality when Local Audit Storage Space is full (whichever is supported by the TOE) are performed. The relevant information is found in the following section(s): **TSF activity** | **AGD Section** determines or modifies the behaviour of transmitting audit data to an external IT entity | Audit Server Configuration handling of audit data, audit functionality when Local Audit Storage Space is full | System Behavior Upon investigation, the evaluator found that the required administrative configuration settings related to TSF data is included in the AGD.

Verdict:
PASS.

5.2.3.2 FMT_MTD.1/CRYPTOKEYS MANAGEMENT OF TSF DATA

5.2.3.2.1 FMT_MTD.1/CRYPTOKEYS TSS

Evaluator Findings:
The TOE is not distributed; hence, this requirement is not applicable.

For non-distributed TOEs, the evaluator shall ensure the TSS lists the keys the Security Administrator is able to manage to include the options available (e.g. generating keys, importing keys, modifying keys or deleting keys) and how that how those operations are performed.

Evaluator Findings:
The evaluator examined the TSS and ensured that, for non-distributed TOEs, it lists the keys the Security Administrator is able to manage to include the options available (e.g. generating keys, importing keys, modifying keys or deleting keys) and how that how those operations are performed. The relevant information is found in the following section(s): **FMT_MTD.1/CryptoKeys** Upon investigation, the evaluator found that: The Cryptographic keys the TOE uses together with their storage and method of destruction are listed in Table 16 of the ST. **Management of cryptographic keys is through the CLI and WebUI[25] as part of managing and configuring SSHv2 and TLS. All key management operations occur through the CLI as well as WebUI commands.**

Verdict:
PASS.

5.2.3.2.2 FMT_MTD.1/CRYPTOKEYS AGD

For distributed TOEs see Section 2.4.1.2.

Evaluator Findings:
The TOE is not distributed; hence, this requirement is not applicable.

For non-distributed TOEs, the evaluator shall also ensure the Guidance Documentation lists the keys the Security Administrator is able to manage to include the options available (e.g. generating keys, importing keys, modifying keys or deleting keys) and how that how those operations are performed.

[25] Only VX series models doesn't support Web UI Feature.

Evaluator Findings:
The evaluator examined the AGD and ensured that, for non-distributed TOEs, it lists the keys the Security Administrator is able to manage to include the options available (e.g. generating keys, importing keys, modifying keys or deleting keys) and how that how those operations are performed. The relevant information is found in the following section(s): **Configuring X.509 certificate Authentication for the Web UI**[26] and **Configuring SSH Public Keys** Upon investigation, the evaluator found that the AGD states that: Use the commands in this section to create a new host key for SSH user authentication: **To configure minimum key length, following command is required:** **'hostname (config) # ssh server min-key-length <key length>'** **To generate server Host Key, following command is required:** **'hostname (config) # ssh server host-key generate'** **To configure the TOE to support RSA based SSH authentication method.** **'SSH server host-key <rsa2> public-key '<public key generated by server>'** **To issue a certificate signing request (CSR), the following command must be executed,** **'hostname (config) # crypto certificate signing-request generate'** **To delete a certificate signing request (CSR), the following command must be executed,** **'hostname (config) # no crypto certificate signing-request csr-name XXX'**

6 SECURITY ASSURANCE REQUIREMENTS

6.1 ASE: SECURITY TARGET EVALUATION

6.1.1 GENERAL ASE

6.1.1.1 EVALUATION ACTIVITY

When evaluating a Security Target, the evaluator performs the work units as presented in the CEM. In addition, the evaluator ensures the content of the TSS in the ST satisfies the EAs specified in Section 2 (Evaluation Activities for SFRs).

Evaluator Findings:
The evaluator performed the work units as presented in the CEM. Furthermore, the evaluator examined the TTS and ensured the content of the TSS in the ST satisfies the EAs specified in Section 2 (Evaluation Activities for SFRs).

[26] VX series models don't support Web UI Fetaure

Verdict:
PASS.

For distributed TOEs only the SFRs classified as 'all' have to be fulfilled by all TOE parts. The SFRs classified as 'One' or 'Feature Dependent' only have to be fulfilled by either one or some TOE parts, respectively. To make sure that the distributed TOE as a whole fulfills all the SFRs the following actions for ASE_TSS.1 have to be performed as part of ASE_TSS.1.1E.

Evaluator Findings:
The TOE is not a distributed TOE; hence this assurance activity is not applicable.

6.2 ADV: DEVELOPMENT

6.2.1 BASIC FUNCTIONAL SPECIFICATION (ADV_FSP.1)

6.2.1.1 EVALUATION ACTIVITY

The evaluator shall examine the interface documentation to ensure it describes the purpose and method of use for each TSFI that is identified as being security relevant.

In this context, TSFI are deemed security relevant if they are used by the administrator to configure the TOE, or to perform other administrative functions (e.g. audit review or performing updates). Additionally, those interfaces that are identified in the ST, or guidance documentation, as adhering to the security policies (as presented in the SFRs), are also considered security relevant. The intent is that these interfaces will be adequately tested and having an understanding of how these interfaces are used in the TOE is necessary to ensure proper test coverage is applied.

The set of TSFI that are provided as evaluation evidence are contained in the Administrative Guidance and User Guidance.

Evaluator Findings:
TOE Design information that can be made public is available in the guidance documentation and in the ST. Any sensitive or proprietary information required by this protection profile is not to be made public. It is not necessary to provide a complete specification of the TSFIs. For NDcPP, additional "functional specification" documentation is not necessary because this requirement is satisfied by multiple other documents (AGD, TSS, and Testing). All associated activities are covered in the Test Report, ST, and AGD documents. NDcPP2.2e, section 7.2.1 states that: "For this cPP, the Evaluation Activities for this family focus on understanding the interfaces presented in the TSS in response to the functional requirements and the interfaces presented in the AGD documentation."

All of the above information is applicable to the ADV Evaluation Activities (5.2.1.1, 5.2.1.2, and 5.2.1.3) in NDcPP2.2e-SD.

The evaluator examined the ST (Security Target) and the AGD (interface documentation) to verify that it describes the purpose and method of use for each TSFI that is identified as being security relevant. The evaluator examined the entire AGD. The evaluator verified the AGD describes the purpose and method of use for each security relevant TSFI by verifying the AGD satisfies all the AGD Evaluation Activities.

During testing, the evaluator used the product and its interfaces extensively and did not find any areas that were deficient.

Verdict:
PASS.

6.2.1.2 EVALUATION ACTIVITY

The evaluator shall check the interface documentation to ensure it identifies and describes the parameters for each TSFI that is identified as being security relevant.

Evaluator Findings:
The evaluator checked the interface documentation (AGD) and ensured it identifies and describes the parameters for each TSFI that is identified as being security relevant. This is covered in the previous evaluation activity above.

Verdict:
PASS.

6.2.1.3 EVALUATION ACTIVITY

The evaluator shall examine the interface documentation to develop a mapping of the interfaces to SFRs.

The evaluator uses the provided documentation and first identifies, and then examines a representative set of interfaces to perform the EAs presented in Section 2, including the EAs associated with testing of the interfaces.

It should be noted that there may be some SFRs that do not have an interface that is explicitly “mapped” to invoke the desired functionality. For example, generating a random bit string, destroying a cryptographic key that is no longer needed, or the TSF failing to a secure state, are capabilities that may be specified in SFRs, but are not invoked by an interface.

However, if the evaluator is unable to perform some other required EA because there is insufficient design and interface information, then the evaluator is entitled to conclude that an adequate functional specification has not been provided, and hence that the verdict for the ADV_FSP.1 assurance component is a ‘fail’.

Evaluator Findings:
The evaluator examined the interface documentation to develop a mapping of the interfaces to SFRs. The evaluator used the provided documentation to first identify, and then examine a representative set of interfaces to perform the EAs presented in Section 2, including the EAs associated with testing of the interfaces. This is covered in the previous evaluation activity above.

Verdict:
PASS.

6.3 AGD: GUIDANCE DOCUMENTS

6.3.1 OPERATIONAL USER GUIDANCE (AGD_OPE.1)

6.3.1.1 EVALUATION ACTIVITY

The evaluator shall ensure the Operational guidance documentation is distributed to Security Administrators and users (as appropriate) as part of the TOE, so that there is a reasonable guarantee that Security Administrators and users are aware of the existence and role of the documentation in establishing and maintaining the evaluated configuration.

Evaluator Findings:
The evaluator checked the requirements above are met by the AGD. The AGD is distributed to administrators and users (as appropriate) as part of the TOE, so that there is a reasonable guarantee that administrators and users are aware of the existence and role of the documentation in establishing and maintaining the evaluated configuration. Upon investigation, the evaluator found that the CC guidance will be published with the CC certificate on www.niap-ccevs.org.

Verdict:
PASS.

6.3.1.2 EVALUATION ACTIVITY

The evaluator shall ensure that the Operational guidance is provided for every Operational Environment that the product supports as claimed in the Security Target and shall adequately address all platforms claimed for the TOE in the Security Target.

Evaluator Findings:
The evaluator ensured that the AGD is provided for every Operational Environment that the product supports as claimed in the Security Target. The section titled **Supported Platforms** of the AGD was used to determine the verdict of this assurance activity. The AGD specifies that the platforms supported are: • AX5600 • CM4600 • CM7600

- CM9600
- EX3600
- EX5600
- EX8600
- FX6600
- HX4600
- NX2600
- NX3600
- NX4600
- NX5600
- NX6600
- NX8600
- VX5600
- VX12600
- CM1500V
- CM2500V
- CM7500V
- EX5500V
- FX2500V
- HX4502V
- HX4600V
- NX1500V
- NX2500V
- NX2550V
- NX4500V
- NX6500V
- NX7500V
- NX8500V
- NX10500V

Verdict:
PASS.

6.3.1.3 EVALUATION ACTIVITY

The evaluator shall ensure that the Operational guidance contains instructions for configuring any cryptographic engine associated with the evaluated configuration of the TOE. It shall provide a warning to the administrator that use of other cryptographic engines was not evaluated nor tested during the CC evaluation of the TOE.

Evaluator Findings:
The evaluator ensured that the AGD contains instructions for configuring any cryptographic implementation associated with the evaluated configuration of the TOE. It provides a warning to the administrator that use of other cryptographic implementations was not evaluated nor tested during the CC evaluation of the TOE.

Verdict:
PASS.

6.3.1.4 EVALUATION ACTIVITY

The evaluator shall ensure the Operational guidance makes it clear to an administrator which security functionality and interfaces have been assessed and tested by the EAs.

Evaluator Findings:
The entire AGD was used to determine the verdict of this work unit. Each confirmation command indicates tested options. Additionally, the section titled **Operational Environment** specifies features that are not assessed and tested by the EAs. The evaluator ensured the AGD makes it clear to an administrator which security functionality and interfaces have been assessed and tested by the EAs.

Verdict:
PASS.

6.3.1.5 EVALUATION ACTIVITY **[TD0536]**

In addition, the evaluator shall ensure that the following requirements are also met:

- The AGD shall contain instructions for configuring any cryptographic engine associated with the evaluated configuration of the TOE. It shall provide a warning to the administrator that use of other cryptographic engines was not evaluated nor tested during the CC evaluation of the TOE.

- **[TD0536]** The documentation must describe the process for verifying updates to the TOE for each method selected for FPT_TUD_EXT.1.3 in the Security Target. The evaluator shall verify that this process includes the following steps:
    - Instructions for obtaining the update itself. This should include instructions for making the update accessible to the TOE (e.g., placement in a specific directory).
    - Instructions for initiating the update process, as well as discerning whether the process was successful or unsuccessful. This includes instructions that describe at least one method of validating the hash/digital signature.

- The TOE will likely contain security functionality that does not fall in the scope of evaluation under this cPP. The AGD shall make it clear to an administrator which security functionality is covered by the Evaluation Activities.

Evaluator Findings:
The evaluator verified the AGD contains instructions for configuring any cryptographic implementations in the section **Cryptographic Protocols** and found that AGD states that **the Enabling**

CC-NDcPP compliance ensures that only certified algorithms and key sizes are available for use by the appliance.

The evaluator verified the AGD contains instructions for initiating the update process, as well as discerning whether the process was successful or unsuccessful.

The relevant information is found in the following section(s): **Software Updates**

Upon investigation, the evaluator found that the AGD states that:

To perform a software update, query the currently active version and view installation status (allows the administrator to see the installed but inactive version). Use the following commands to install new software images,

- **Download the software image:**

hostname (config) # image fetch <location of image>

- **View download progress:**

hostname (config) # show <location of image> image status

- **To verify the version of downloaded image:**

hostname (config) # show images

- **Install the downloaded software image:**

hostname (config) # image install <image-lms_7.9.0.img>

hostname (config) # image boot next

- **Save changes:**

hostname (config) # reload

- **Show software version:**

hostname (config) # show version

Software image files are digitally signed so their integrity can be automatically verified during the upgrade process. An image that fails an integrity check will not be loaded.

The evaluator verified that the AGD contains details about the security functionality of the TOE in the section **Supported Platforms** and found that AGD states that: **Each model of the TOE shares an identical codebase employing all NDcPP required functionality. Breach detection, email analysis, endpoint monitoring, IPS, malware analysis, and threat prevention features are not evaluated as part of the Common Criteria certification and are excluded from the evaluated configuration.**

Verdict:
PASS.

6.3.2 PREPARATIVE PROCEDURES (AGD_PRE.1)

6.3.2.1 EVALUATION ACTIVITY

The evaluator shall examine the Preparative procedures to ensure they include a description of how the Security Administrator verifies that the operational environment can fulfil its role to support the security functionality (including the requirements of the Security Objectives for the Operational Environment specified in the Security Target).

The documentation should be in an informal style and should be written with sufficient detail and explanation that they can be understood and used by the target audience (which will typically include IT staff who have general IT experience but not necessarily experience with the TOE product itself).

Evaluator Findings:
The evaluator examined the Preparative procedures to ensure they include a description of how the administrator verifies that the operational environment can fulfil its role to support the security functionality. The evaluator reviewed the sections titled **Operational Environment** of the AGD. The evaluator found that these sections describe how the Operational Environment must meet: • Virtual Hardware • Management Workstation with Web Browser and SSH Client • Syslog server • NTP Server

Verdict:
PASS.

6.3.2.2 EVALUATION ACTIVITY

The evaluator shall examine the Preparative procedures to ensure they are provided for every Operational Environment that the product supports as claimed in the Security Target and shall adequately address all platforms claimed for the TOE in the Security Target.

Evaluator Findings:

The evaluator checked the requirements above are met by the preparative procedures. The entire AGD was used to determine the verdict of this work unit. Upon investigation, the evaluator found that the AGD describes each of the devices in the operating environment, including,

Component	Usage/Purpose Description for TOE performance
Virtual Hardware	Virtual hardware provided by VMware vSphere ESXi 7.0 and Intel(R) Xeon(R) CPU E5-4620 v4(Broadwell)
Management Workstation with Web Browser and SSH Client	This includes any IT Environment Management workstation with a Web Browser and a SSH client installed that is used by the TOE administrator to support TOE administration through HTTPS[27] and

[27] VX series models don't support Web UI Fetaure

	SSH protected channels. Any SSH client that supports SSHv2 may be used. Any web browser that supports TLS 1.2 may be used.
Audit server	The syslog audit server is used for remote storage of audit records that have been generated by and transmitted from the TOE. The syslog server must support communications using TLS 1.2.
NTP Server	NTP server supporting SHA-1 integrity verification.

The section titled **Operational Environment** of AGD identifies the following supported platform:

Category	Identifier
Physical Appliances	AX5600 CM4600 CM7600 CM9600 EX3600 EX5600 EX8600 FX6600 HX4600 NX2600 NX3600 NX4600 NX5600 NX6600 NX8600 VX5600 VX12600
Virtual Appliances	CM1500V CM2500V CM7500V EX5500V FX2500V HX4502V HX4600V NX1500V NX2500V NX2550V NX4500V NX6500V NX7500V NX8500V NX10500V
Software Version	TRFEOS 10.0.4

Verdict:
PASS.

6.3.2.3 EVALUATION ACTIVITY

The evaluator shall examine the preparative procedures to ensure they include instructions to successfully install the TSF in each Operational Environment.

Evaluator Findings:

The evaluator checked the requirements are met by the preparative procedures. The entire AGD was used to determine the verdict of this work unit. Upon investigation, the evaluator found that AGD describes all of the instructions necessary to install and configure the TOE to work in the target operating environment, including:

- Configuring Administrative Accounts and Passwords
- Configuring SSH
- Configuring TLS
- Configuring the Remote Syslog Server
- Configuring Audit Log Options
- Configuring Event Logging
- Configuring a Secure Logging Channel
- Configuring Software Updates
- Configuring Setting Time
- Configuring Login Banners

Verdict:
PASS.

6.3.2.4 EVALUATION ACTIVITY

The evaluator shall examine the preparative procedures to ensure they include instructions to manage the security of the TSF as a product and as a component of the larger operational environment.

Evaluator Findings:
The evaluator ensured the preparative procedures include instructions to manage the security of the TSF as a product and as a component of the larger operational environment. The entire AGD was used to determine the verdict of this work unit. The same commands, configurations, and interfaces used to install the TOE are also used for ongoing management, so this is satisfied by AGD_PRE.1 Test #3

Verdict:
PASS.

6.3.2.5 EVALUATION ACTIVITY

In addition, the evaluator shall ensure that the following requirements are also met. The preparative procedures must:

a) include instructions to provide a protected administrative capability; and

b) identify TOE passwords that have default values associated with them and instructions shall be provided for how these can be changed.

Evaluator Findings:
The evaluator ensured the preparative procedures include instructions to provide a protected administrative capability and changing default passwords. The sections titled **TOE Administration** were used to determine the verdict of this work unit. The AGD describes all the instructions necessary to provide protected administrative capability, including the following: • Configuring the number of failed attempts or lockout time • Resetting passwords • Creating new users with strong passwords

Verdict:
PASS.

6.4 AVA: VULNERABILITY ASSESSMENT

6.4.1 VULNERABILITY SURVEY (AVA_VAN.1)

6.4.1.1 EVALUATION ACTIVITY (DOCUMENTATION) **[TD0547]**

In addition to the activities specified by the CEM in accordance with Table 2, the evaluator shall perform the following activities.

The evaluator shall examine the documentation outlined below provided by the developer to confirm that it contains all required information. This documentation is in addition to the documentation already required to be supplied in response to the EAs listed previously.

The developer shall provide documentation identifying the list of software and hardware components that compose the TOE. Hardware components should identify at a minimum the processors used by the TOE. Software components include applications, the operating system and other major components that are independently identifiable and reusable (outside of the TOE), for example a web server, protocol or cryptographic libraries, (independently identifiable and reusable components are not limited to the list provided in the example). This additional documentation is merely a list of the name and version number of the components and will be used by the evaluators in formulating vulnerability hypotheses during their analysis.
If the TOE is a distributed TOE then the developer shall provide:

a. documentation describing the allocation of requirements between distributed TOE components as in [NDcPP, 3.4]
b. a mapping of the auditable events recorded by each distributed TOE component as in [NDcPP, 6.3.3]
c. additional information in the Preparative Procedures as identified in the refinement of AGD_PRE.1 in additional information in the Preparative Procedures as identified in 3.4.1.2 and 3.5.1.2.

Evaluator Findings:
The evaluator collected this information from the developer which was used to feed into the Public Domain Search. Refer to the evaluator findings in the evaluation activity below.

Verdict:
PASS.

6.4.1.2 EVALUATION ACTIVITY [**TD0564 APPLIED**] [**LABGRAM #116**]

The evaluator formulates hypotheses in accordance with process defined in Appendix A. The evaluator documents the flaw hypotheses generated for the TOE in the report in accordance with the guidelines in Appendix A.3. The evaluator shall perform vulnerability analysis in accordance with Appendix A.2. The results of the analysis shall be documented in the report according to Appendix A.3.

Evaluator Findings:
The evaluator documented their analysis and testing of potential vulnerabilities with respect to this requirement. Public searches were performed against all keywords found within the Security Target and AGD that may be applicable to specific TOE components. This included protocols, TOE software version, and TOE hardware to ensure sufficient coverage under AVA. The evaluator searched the Internet for potential vulnerabilities in the TOE using the web sites listed below. The sources of the publicly available information are provided below:

- http://nvd.nist.gov/
- http://www.us-cert.gov
- http://www.securityfocus.com/
- https://www.cvedetails.com/
- www.exploitsearch.net
- www.securiteam.com
- http://nessus.org/plugins/index.php?view=search
- http://www.zerodayinitiative.com/advisories
- https://www.exploit-db.com
- https://www.rapid7.com/db/vulnerabilities

The evaluator performed the public domain vulnerability searches using the following key words. The search was performed on September 26, 2024:

- FireEye
- Trellix
- TRFEOS 10.0.4
- AX5600
- CM4600
- CM7600
- CM9600
- EX3600
- EX5600
- EX8600
- FX6600
- HX4600
- NX2600
- NX3600
- NX4600
- NX5600
- NX6600
- NX8600
- VX5600
- VX12600
- CM1500V
- CM2500V
- CM7500V
- EX5500V
- FX2500V
- HX4502V
- HX4600V
- NX1500V
- NX2500V
- NX2550V
- NX4500V
- NX6500V

- NX7500V
- NX8500V
- NX10500V
- Intel Xeon E-2334 (Rocket Lake)
- Intel Xeon Silver 4314 (Ice Lake)
- Intel Xeon Silver 4316 (Ice Lake)
- Intel Xeon E-2378 (Rocket Lake)
- Intel Xeon Gold 6330 (Ice Lake)
- Intel Xeon Platinum 8380 (Ice Lake)
- Intel Xeon E5-4620 v4 (Broadwell)
- Dell PowerEdge R830
- Trellix OpenSSL FIPS Object Module
- libcrypt.so
- OpenSSH 7.4p1
- Apache 2.4.62 (CentOS Linux)
- OpenSSL 1.0.2zh

The evaluation lab examined each result provided from NVD and Exploit Search to determine if the current TOE version or component within the environment was vulnerable. Based upon the analysis, any issues found that were generated were patched in the TOE version and prior versions, mitigating the risk factor.

Verdict:
PASS.

6.4.1.3 EVALUATION ACTIVITY 2

The evaluator shall perform the following activities to generate type 4 flaw hypotheses:

Fuzz testing

- Examine effects of sending:
  - mutated packets carrying each 'Type' and 'Code' value that is undefined in the relevant RFC for each of ICMPv4 (RFC 792) and ICMPv6 (RFC 4443)
  - mutated packets carrying each 'Transport Layer Protocol' value that is undefined in the respective RFC for IPv4 (RFC 791) IPv6 (RFC 2460) should also be covered if it is supported and claimed by the TOE.
  - Since none of these packets will belong to an allowed session, the packets should not be processed by the TOE, and the TOE should not be adversely affected by this traffic. Any results that are unexpected (e.g., core dumps) are candidates for a flaw hypothesis.
- Mutation fuzz testing of the remaining fields in the required protocol headers. This testing requires sending mutations of well- formed packets that have both carefully chosen and random values inserted into each header field in turn (i.e. testing is to include both carefully chosen and random insertion test cases). The original well-formed packets would be accepted as part of a normal existing communication stream and may still be accepted as valid packets when subject to the carefully chosen mutations (the individual packet alone would be valid although its contents may not be valid in the context of preceding and/or following packets), but will often

not be valid packets when random values are inserted into fields. The carefully chosen values should include semantically significant values that can be determined from the type of the data that the field represents, such as values indicating positive and negative integers, boundary conditions, invalid binary combinations (e.g. for flag sets with dependencies between bits), and missing start or end values. Randomly chosen values may not result in well-formed packets but are included nonetheless to see whether they can lead to the device entering an insecure state. Any results that are unexpected (e.g., core dumps) are candidates for a flaw hypothesis.

Evaluator Findings:
The evaluator documented the fuzz testing results with respect to this requirement. The evaluation lab examined each result from fuzz testing to determine if the TOE improperly processes packets. Based upon the analysis, no unexpected results occurred. Therefore, no Type 4 hypotheses were generated.

Verdict:
PASS.

6.4.1.4 EVALUATION ACTIVITY 3

The following additional tests shall be performed:1.) [Conditional]: If the TOE is a TLS server and supports ciphersuites that use RSA transport (e.g. supporting TLS_RSA_WITH_* ciphers) the following test shall be performed. Where RSA Key Establishment schemes are claimed and especially when PKCS#1 v1.5* padding is used, the evaluators shall test for implementation flaws allowing Bleichenbacher and Klima et al. style attacks, including Bock et al's ROBOT attacks of 2017 in the flaw analysis. Even though Bleichenbacher's original paper is two decades old, Bock et al. found

these attacks to still be effective in weakening the security of RSA key establishment in current products. Bleichenbacher and Klima et al. style attacks are complex and may be difficult to detect, but a number of software testing tools have been created to assist in that process. The iTC strongly recommends that at least one of the tools mentioned in Bock et al's ROBOT attacks of 2017 webpage or paper, as effective to detect padding oracle attacks, be used to test TOE communications channels using RSA based Key

Establishment (related sources:
http://archiv.infsec.ethz.ch/education/fs08/secsem/bleichenbacher98.pdf,

https://eprint.iacr.org/2003/052, https://robotattack.org/). Network Device Equivalency Considerations

Evaluator Findings:
N/A, even though the TOE is a TLS server but does not support ciphersuites that use RSA transport.

Verdict:
PASS.

7 DETAILED TEST CASES (TEST ACTIVITIES)

7.1 AUDIT

7.1.1 FAU_GEN.1 TEST #1

Item	Data
Test Assurance Activity	The evaluator shall test the TOE's ability to correctly generate audit records by having the TOE generate audit records for the events listed in the table of audit events and administrative actions listed above. This should include all instances of an event: for instance, if there are several different I&A mechanisms for a system, the FIA_UIA_EXT.1 events must be generated for each mechanism. The evaluator shall test that audit records are generated for the establishment and termination of a channel for each of the cryptographic protocols contained in the ST. If HTTPS is implemented, the test demonstrating the establishment and termination of a TLS session can be combined with the test for an HTTPS session. When verifying the test results, the evaluator shall ensure the audit records generated during testing match the format specified in the guidance documentation, and that the fields in each audit record have the proper entries. Note that the testing here can be accomplished in conjunction with the testing of the security mechanisms directly.
Test Steps	• Trigger each auditable event on the TOE. Verify that each audit record is generated and contains the required information.
Expected Test Results	• The TOE should accurately generate audit records for all the required auditable events. • Evidence- Snapshot showing generated logs for audit records.
Pass/Fail with Explanation	Pass. The audit records associated with each test case are recorded with each test case. A comparison of required audit records to the presented audit records was additionally performed. This analysis shows that each required audit record is generated by the TOE.

7.1.2 FAU_GEN.2 TEST #1

Item	Data

Test Assurance Activity	This activity should be accomplished in conjunction with the testing of FAU_GEN.1.1.
Pass/Fail with Explanation	Pass. FAU_GEN.1 Test#1 covers this requirement.

7.1.3 FAU_GEN.2 TEST #2

Item	Data
Test Assurance Activity	For distributed TOEs the evaluator shall verify that where auditable events are instigated by another component, the component that records the event associates the event with the identity of the instigator. The evaluator shall perform at least one test on one component where another component instigates an auditable event. The evaluator shall verify that the event is recorded by the component as expected and the event is associated with the instigating component. It is assumed that an event instigated by another component can at least be generated for building up a secure channel between two TOE components. If for some reason (could be e.g. TSS or Guidance Documentation) the evaluator would come to the conclusion that the overall TOE does not generate any events instigated by other components, then this requirement shall be omitted.
Pass/Fail with Explanation	N/A. TOE is not distributed.

7.1.4 FAU_STG_EXT.1 TEST #1

Item	Data
Test Assurance Activity	Testing of the trusted channel mechanism for audit will be performed as specified in the associated assurance activities for the particular trusted channel mechanism. The evaluator shall perform the following additional tests for this requirement: Test 1: The evaluator shall establish a session between the TOE and the audit server according to the configuration guidance provided. The evaluator shall then examine the traffic that passes between the audit server and the TOE during

	several activities of the evaluator's choice designed to generate audit data to be transferred to the audit server. The evaluator shall observe that these data are not able to be viewed in the clear during this transfer, and that they are successfully received by the audit server. The evaluator shall record the particular software (name, version) used on the audit server during testing. The evaluator shall verify that the TOE is capable of transferring audit data to an external audit server automatically without administrator intervention.
Test Steps	• Configure the TOE to send logs to a Syslog server. • Configure the Syslog server with port and certificates. • Verify the Syslog version on the VM. • Restart the Syslog service. • Verify the logs generated on the TOE. • Verify the logs seen on the remote Syslog server are the same. • Verify via packet capture that traffic between TOE and the Syslog server is not sent in plaintext.
Expected Test Results	• The TOE should support the transfer of audit data without admin intervention. • The communication between TOE and Syslog server should be encrypted. • Packet Capture should show that traffic between TOE and the Syslog server is not sent in plaintext. • TOE logs should show a successful Syslog connection.
Pass/Fail with Explanation	Pass. The TOE passes all audit traffic to the remote audit server through a secure channel without admin interference. The evaluator accurately records the specific software used on the audit server, including the name and version. This meets the testing requirements.

7.1.5 FAU_STG_EXT.1 TEST #2 (A)

Item	Data
Test Assurance Activity	Testing of the trusted channel mechanism for audit will be performed as specified in the associated assurance activities for the particular trusted channel mechanism. The evaluator shall perform the following additional tests for this requirement: Test 2: The evaluator shall perform operations that generate audit data and verify that this data is stored locally. The evaluator shall perform operations that generate audit data until the local storage space is exceeded and verifies that the

	TOE complies with the behaviour defined in FAU_STG_EXT.1.3. Depending on the configuration this means that the evaluator has to check the content of the audit data when the audit data is just filled to the maximum and then verifies that: The audit data remains unchanged with every new auditable event that should be tracked but that the audit data is recorded again after the local storage for audit data is cleared (for the option '**drop new audit data**' in FAU_STG_EXT.1.3).
Pass/Fail with Explanation	N/A. The option 'drop new audit data' is not selected in the ST.

7.1.6 FAU_STG_EXT.1 TEST #2 (B)

Item	Data
Test Assurance Activity	Testing of the trusted channel mechanism for audit will be performed as specified in the associated assurance activities for the particular trusted channel mechanism. The evaluator shall perform the following additional tests for this requirement: Test 2: The evaluator shall perform operations that generate audit data and verify that this data is stored locally. The evaluator shall perform operations that generate audit data until the local storage space is exceeded and verifies that the TOE complies with the behaviour defined in FAU_STG_EXT.1.3. Depending on the configuration this means that the evaluator has to check the content of the audit data when the audit data is just filled to the maximum and then verifies that: The existing audit data is overwritten with every new auditable event that should be tracked according to the specified rule (for the option '**overwrite previous audit records**' in FAU_STG_EXT.1.3)
Test Steps	• Configure the smallest possible logging space. • Observe the last archived file's date and time. • Wait for the current log file to reach its limit and be archived. Verify TOE replaced the last archive file.
Expected Test Results	• The TOE should successfully allow the overwriting of old log files by new ones. • Evidence – snapshot should show that the oldest log file is overwritten by the new log file.

Pass/Fail with Explanation	Pass. The test is passed because once the limit was reached the oldest audit record was overwritten. This meets the testing requirements.

7.1.7 FAU_STG_EXT.1 TEST #2 (C)

Item	Data
Test Assurance Activity	Testing of the trusted channel mechanism for audit will be performed as specified in the associated assurance activities for the particular trusted channel mechanism. The evaluator shall perform the following additional tests for this requirement: Test 2: The evaluator shall perform operations that generate audit data and verify that this data is stored locally. The evaluator shall perform operations that generate audit data until the local storage space is exceeded and verifies that the TOE complies with the behaviour defined in FAU_STG_EXT.1.3. Depending on the configuration this means that the evaluator has to check the content of the audit data when the audit data is just filled to the maximum and then verifies that: The TOE behaves as specified (for the option '**other action**' in FAU_STG_EXT.1.3).
Pass/Fail with Explanation	N/A. The option 'other action' is not selected in the ST.

7.1.8 FAU_STG_EXT.1 TEST #3

Item	Data
Test Assurance Activity	Test 3: If the TOE complies with FAU_STG_EXT.2/LocSpace the evaluator shall verify that the numbers provided by the TOE according to the selection for FAU_STG_EXT.2/LocSpace are correct when performing the tests for FAU_STG_EXT.1.3

Pass/Fail with Explanation	N/A. FAU_STG_EXT.2/LocSpace not claimed in ST.

7.1.9 FAU_STG_EXT.1 TEST #4

Item	Data
Test Assurance Activity	Test 4: For distributed TOEs, Test 1 defined above should be applicable to all TOE components that forward audit data to an external audit server. For the local storage according to FAU_STG_EXT.1.2 and FAU_STG_EXT.1.3 the Test 2 specified above shall be applied to all TOE components that store audit data locally. For all TOE components that store audit data locally and comply with FAU_STG_EXT.2/LocSpace Test 3 specified above shall be applied. The evaluator shall verify that the transfer of audit data to an external audit server is implemented.
Pass/Fail with Explanation	N/A. TOE is a standalone.

7.1.10 FCS_NTP_EXT.1.1 TEST #1

Item	Data
Test Assurance Activity	The version of NTP selected in element 1.1 and specified in the ST shall be verified by observing establishment of a connection to an external NTP server known to be using the specified version(s) of NTP. This may be combined with tests of other aspects of FCS_NTP_EXT.1 as described below.
Test Steps	NTP V3 • Configure NTP on the TOE. • Verify NTP on TOE. • Verify the NTP version with packet capture.

	• Verify the NTP version via logs and ensure that the time is set through the added NTP server. NTP V4 • Configure NTP on the TOE. • Verify NTP on TOE. • Verify the NTP version with packet capture. • Verify the NTP version via logs and ensure that the time is set through the added NTP server.
Expected Test Results	• When the NTP server is configured, the TOE should successfully establish a connection with the configured NTP server. • TOE should sync with the configured version (V3/V4) of NTP. • Packet capture and device logs should show a successful connection with the NTP server.
Pass/Fail with Explanation	Pass. The test passes as the TOE is successfully able to synchronize the time with the NTP server using the NTP version selected in element 1.1 and specified in the ST.

7.1.11 FCS_NTP_EXT.1.2 TEST #1

Item	Data
Test Assurance Activity	[Conditional] If the **message digest algorithm** is claimed in element 1.2, the evaluator will change the message digest algorithm used by the NTP server in such a way that the new value does not match the configuration on the TOE and confirms that the TOE does not synchronize to this time source. The evaluator shall use a packet sniffer to capture the network traffic between the TOE and the NTP server. The evaluator uses the captured network traffic, to verify the NTP version, to observe time change of the TOE and uses the TOE's audit log to determine that the TOE accepted the NTP server's timestamp update. The captured traffic is also used to verify that the appropriate message digest algorithm was used to authenticate the time source and/or the appropriate protocol was used to ensure integrity of the timestamp that was transmitted in the NTP packets.

Test Steps	NTP V3 • Configure and enable SHA1 authentication for NTP server version 3. • Configure the NTP server with a supported message digest algorithm by the TOE. • Verify that the NTP synchronization succeeds. • Verify the successful NTP synchronization via packet capture. • Verify that the time is synchronized with the NTP server via logs. • Modify the message digest algorithm used by the NTP server. • Verify that the NTP synchronization fails. • Verify that the NTP synchronization has failed via packet capture. • Verify that the time synchronization has failed via TOE logs. NTPV4 • Configure and enable SHA1 authentication for NTP server version 4. • Configure the NTP server with a supported message digest algorithm by the TOE. • Verify that the NTP synchronization succeeds. • Verify the successful NTP synchronization via packet capture. • Verify that the time is synchronized with the NTP server via logs. • Modify the message digest algorithm used by the NTP server. • Verify that the NTP synchronization fails. • Verify that the NTP synchronization has failed via packet capture. • Verify that the time synchronization has failed via TOE logs.
Expected Test Results	• When the NTP server (version 3 and version 4) with a supported message-digest algorithm is configured, synchronization between TOE and NTP server should succeed. • TOE logs and Packet capture should show a successful connection due to a supported message digest algorithm. • When the NTP server (version 3 and version 4) with an unsupported message-digest algorithm is configured, synchronization between TOE and NTP server should fail. • TOE logs and Packet capture should show connection failure due to an unsupported message digest algorithm.
Pass/Fail with Explanation	Pass. The TOE syncs with the NTP Server with version 3 and version 4 when the supported message-digest algorithm is configured and does not sync when an unsupported message digest algorithm is used, this meets testing requirements.

7.1.12 FCS_NTP_EXT.1.3 TEST #1

Item	Data
Test Assurance Activity	The evaluator shall configure NTP server(s) to support periodic time updates to broadcast and multicast addresses. The evaluator shall confirm the TOE is configured to not accept broadcast and multicast NTP packets that would result in the timestamp being updated. The evaluator shall check that the time stamp is not updated after receipt of the broadcast and multicast packets.
Test Steps	Broadcast: • Check the time on the TOE. • Set the NTP server to broadcast to 10.1.3.255. • Verify with a packet capture that broadcast packets are sent by the NTP server. • Verify that the time on the TOE is not updated by NTP. Multicast: • Check the time on the TOE. • Set the NTP server to multicast to 224.0.1.1. • Verify with a packet capture that multicast packets are sent by the NTP server. • Check the time on TOE and verify that it is not updated by NTP.
Expected Test Results	• TOE should not accept broadcast and multicast updates from the NTP server. • Packet capture should show broadcast and multicast packets. • Snapshot should show time on the TOE is not updated.
Pass/Fail with Explanation	Pass. The TOE appropriately rejects any time updates from broadcast or multicast NTP packets. This meets testing requirements.

7.1.13 FCS_NTP_EXT.1.4 TEST #1 [TD0528]

Item	Data
Test Assurance Activity	Test 1: The evaluator shall confirm the TOE supports configuration of at least three (3) NTP time sources. The evaluator shall configure at least three NTP servers to support periodic time updates to the TOE. The evaluator shall confirm the TOE is configured to accept NTP packets that would result in the timestamp being updated from each of the NTP servers. The evaluator shall check that the time stamp is updated after receipt of the NTP packets. The purpose of this test to verify that the TOE can be configured to synchronize with multiple NTP servers. It is up to the evaluator to determine that the multi- source update of the time information is appropriate and consistent with the behavior prescribed by the RFC 1305 for NTPv3 and RFC 5905 for NTPv4.
Test Steps	Version 3 • Verify the current time on the TOE. • Configure at least 3 NTP time sources with version 3. • Verify the NTP configuration on the TOE. • Verify logs are generated for the addition of NTP servers. • Verify to which NTP server TOE is synchronized first. • Verify that the time is synchronized through the first NTP server using TOE logs. • Verify that the time is synchronized through the first NTP server using packet capture. • Verify the updated time on the TOE. • Update the time according to the second NTP server and verify that it synchronizes successfully. • Verify that the time is synchronized through the second NTP server using TOE logs. • Verify that the time is synchronized through the second NTP server using packet capture.

	• Update the time according to the third NTP server and verify that it synchronizes successfully. • Verify that the time is synchronized through the third NTP server using TOE logs. • Verify that the time is synchronized through the third NTP server using packet capture. Version 4 • Verify the current time on the TOE. • Configure at least 3 NTP time sources with version 4. • Verify the NTP configuration on the TOE. • Verify logs are generated for the addition of NTP servers. • Verify to which NTP server TOE is synchronized first. • Verify that the time is synchronized through the first NTP server using TOE logs. • Verify that the time is synchronized through the first NTP server using packet capture. • Verify the updated time on the TOE. • Update the time according to the second NTP server and verify that it synchronizes successfully. • Verify that the time is synchronized through the second NTP server using TOE logs. • Verify that the time is synchronized through the second NTP server using packet capture. • Update the time according to the third NTP server and verify that it synchronizes successfully. • Verify that the time is synchronized through the third NTP server using TOE logs.

	• Verify that the time is synchronized through the third NTP server using packet capture.
Expected Test Results	• The TOE should support the configuration of three NTP servers. • When three NTP servers are configured on the TOE, the TOE should successfully synchronize with all the NTP servers. • Packet captures should show NTP packets are received from each of the NTP servers. • TOE logs should show the addition of NTP servers and time synchronization with them for both NTP version 3 and NTP version 4.
Pass/Fail with Explanation	Pass. The TOE is able to successfully sync its time with multiple configured NTP servers. . This meets testing requirements.

7.1.14 FCS_NTP_EXT.1.4 TEST #2 [**TD0528**]

Item	Data
Test Assurance Activity	Test 2: (The intent of this test is to ensure that the TOE would only accept NTP updates from configured NTP Servers). The evaluator shall confirm that the TOE would not synchronize to other, not explicitly configured time sources by sending an otherwise valid but unsolicited NTP Server responses indicating different time from the TOE's current system time. This rogue time source needs to be configured in a way (e.g. degrade or disable valid and configured NTP servers) that could plausibly result in unsolicited updates becoming a preferred time source if they are not discarded by the TOE. The TOE is not mandated to respond in a detectable way or audit the occurrence of such unsolicited updates. The intent of this test is to ensure that the TOE would only accept NTP updates from configured NTP Servers. It is up to the evaluator to craft and transmit unsolicited updates in a way that would be consistent with the behaviour of a correctly functioning NTP server.
Test Steps	• Verify the time on the TOE. • Configure an NTP server. • Sync the TOE with the NTP server. • Verify that the TOE successfully synced with the configured NTP server using packet capture. • Configure a different NTP server to which the TOE syncs. • Replay the packets from the NTP server which are captured during earlier sync. • Verify the TOE does not sync with the rogue NTP server.
Expected Output	• The timestamp on the TOE should not be updated by an unconfigured or rogue NTP server.

	• Rogue packets should not have an effect on the TOE and TOE should notrespond to them or update the timestamp. • Packet capture should show rouge packets. • Screenshot should show that the time on the TOE is not affected by the rogue packets.
Pass/Fail with Explanation	Pass. The TOE only accepts NTP updates from configured NTP Servers. This meets the testing requirements.

7.1.15 FPT_STM_EXT.1 TEST #1

Item	**Data**
Test Assurance Activity	Test 1: If the TOE supports direct **setting of the time by the Security Administrator** then the evaluator uses the guidance documentation to set the time. The evaluator shall then use an available interface to observe that the time was set correctly.
Test Steps	Console: • Confirm the current time. • Set a new time. • Verify that the time on the TOE was updated. • Verify logs were generated for the time change. SSH: • Confirm the current time. • Set a new time. • Verify that the time on the TOE was updated. • Verify logs were generated for the time change. GUI: • Confirm the current time. • Set a new time. • Verify that the time on the TOE was updated. • Verify logs were generated for the time change.
Expected Test Results	• The TOE should allow time to be set manually over SSH, local console and GUI. • Evidence: Snapshot should show updated time. • TOE should generate logs for the time change.

Pass/Fail with Explanation	Pass. Security Admin is able to set time manually over SSH, local console and GUI on TOE device.. This meets the testing requirement.

7.1.16 FPT_STM_EXT.1 TEST #2

Item	Data
Test Assurance Activity	Test 2: If the TOE supports the **use of an NTP server**; the evaluator shall use the guidance documentation to configure the NTP client on the TOE, and set up a communication path with the NTP server. The evaluator will observe that the NTP server has set the time to what is expected. If the TOE supports multiple protocols for establishing a connection with the NTP server, the evaluator shall perform this test using each supported protocol claimed in the guidance documentation.
Test Steps	NTP V3 • Configure NTP on the TOE. • Verify NTP on TOE. • Verify the NTP version and successful NTP synchronization with packet capture. • Verify the NTP version via logs and ensure that the time is set through the added NTP server. NTP V4 • Configure NTP on the TOE. • Verify NTP on TOE. • Verify the NTP version and successful NTP synchronization with packet capture. • Verify the NTP version via logs and ensure that the time is set through the added NTP server.
Expected Test Results	• TOE should successfully synchronize with the NTP server. • Packet capture and TOE log should show successful synchronization with the NTP server.
Pass/Fail with Explanation	Pass. The TOE was successfully able to synchronize with the NTP server Version v3 and v4. This meets the testing requirements.

7.1.17 FPT_STM_EXT.1 TEST #3 [**TD0632**]

Item	Data
Test Assurance Activity	Test 3: [conditional] If the TOE obtains time from the underlying VS, the evaluator shall record the time on the TOE, modify the time on the underlying VS, and verify the modified time is reflected by the TOE. If there is a delay between the setting the time on the VS and when the time is reflected on the TOE, the evaluator shall ensure this delay is consistent with the TSS and Guidance.
Pass/Fail with Explanation	N/A. The TOE does not obtain time from the underlying VS.

7.1.18 FTP_ITC.1 TEST #1

Item	Data
Test Assurance Activity	The evaluators shall ensure that communications using each protocol with each authorized IT entity is tested during the course of the evaluation, setting up the connections as described in the guidance documentation and ensuring that communication is successful.
Pass/Fail with Explanation	Pass. The test has been exercised in FAU_STG_EXT.1 Test#1 for setting up successful communication with the audit server over TLS.

7.1.19 FTP_ITC.1 TEST #2

Item	Data
Test Assurance Activity	For each protocol that the TOE can initiate as defined in the requirement, the evaluator shall follow the guidance documentation to ensure that in fact the communication channel can be initiated from the TOE.

Test Steps	FAU_STG_EXT.1 Test #1 and FCS_TLSC_EXT.1 cover this test requirement
Pass/Fail with Explanation	Pass. The test has been exercised in FAU_STG_EXT.1 Test#1 using the TLS protocol to protect audit data to an audit server where the TOE is initiating the connection.

7.1.20 FTP_ITC.1 TEST #3

Item	Data
Test Assurance Activity	The evaluator shall ensure, for each communication channel with an authorized IT entity, the channel data is not sent in plaintext.
Test Steps	FAU_STG_EXT.1 Test #1 and FCS_TLSC_EXT.1 cover this test requirements
Pass/Fail with Explanation	Pass. The test has been exercised in FAU_STG_EXT.1 Test#1 and FCS_TLSC_EXT.1.1 Test#1 where a successful TLS connection is established with the audit server and the channel data is encrypted.

7.1.21 FTP_ITC.1 TEST #4

Item	Data
Test Assurance Activity	Objective: The objective of this test is to ensure that the TOE reacts appropriately to any connection outage or interruption of the route to the external IT entities. The evaluator shall, for each instance where the TOE acts as a client utilizing a secure communication mechanism with a distinct IT entity, physically interrupt the connection of that IT entity for the following durations: 1. A duration that exceeds the TOE's application layer timeout setting, 2. A duration shorter than the application layer timeout but of sufficient length to interrupt the network link layer.

	The evaluator shall ensure that, when the physical connectivity is restored, communications are appropriately protected and no TSF data is sent in plaintext. In the case where the TOE is able to detect when the cable is removed from the device, another physical network device (e.g. a core switch) shall be used to interrupt the connection between the TOE and the distinct IT entity. The interruption shall not be performed at the virtual node (e.g. virtual switch) and must be physical in nature.
Test Steps	Short duration: • Establish a connection with the TOE over TLS and verify the successful connection. • Physically disrupt the connection for a short time (duration shorter than the application layer timeout), then test the connection. No data will go through, when connectivity is restored, application data remains encrypted. • Check audit logs for a successful connection with the Syslog server. Long duration: • Establish a connection with the TOE over TLS and verify the successful connection. • Physically disrupt the connection for a long time (duration exceeds application layer timeout), then test the connection. No data will go through. • Verify connection disruption using a packet capture. • When connectivity is restored, the application data remains encrypted. • Check audit logs for a successful connection with the Syslog server.
Expected Test Results	• The data should continue to be encrypted after the connection is restored regardless of the duration. • Evidence - Packet capture should show connection reset and encrypted application data. • TOE log should show logs for successful connection and restored connection.
Pass/Fail with Explanation	Pass. The TOE does not send plaintext traffic when disconnected for a short and long period of time from the Syslog server. This meets the testing requirements.

7.2 AUTH

7.2.1 FCS_HTTPS_EXT.1 TEST #1[28]

Item	Data
Test Assurance Activity	This test is now performed as part of FIA_X509_EXT.1/Rev testing. Tests are performed in conjunction with the TLS evaluation activities. If the TOE is an HTTPS client or an HTTPS server utilizing X.509 client authentication, then the certificate validity shall be tested in accordance with testing performed for FIA_X509_EXT.1.
Pass/Fail with Explanation	Pass. This test is performed as part of FIA_X509_EXT.1/Rev testing. Tests are performed in conjunction with the TLS evaluation activities.

7.2.2 FIA_AFL.1 TEST #1 [**TD0570**]

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests for each method by which remote administrators access the TOE (e.g. any passwords entered as part of establishing the connection protocol or the remote administrator application): Test 1: The evaluator shall use the operational guidance to configure the number of successive unsuccessful authentication attempts allowed by the TOE (and, if the time period selection in FIA_AFL.1.2 is included in the ST, then the evaluator shall also use the operational guidance to configure the time period after which access is re-enabled). The evaluator shall test that once the authentication attempts limit is reached, authentication attempts with valid credentials are no longer successful.
Test Steps	• Configure the maximum number of unsuccessful authentication attempts to 3. • Confirm the configuration has been implemented in the config.

[28]VX series models doesn't support Web UI Feature and hence this selection-based SFR is not applicable to the VX Series Models

	SSH: • Attempt to log in unsuccessfully three times, triggering the lock out. • Attempt to log in a fourth time using the correct credentials. This will fail. • Verify the logs on TOE showing an account is locked out. HTTPS: • Attempt to log in unsuccessfully three times, triggering the lock out. • Attempt to log in a fourth time using the correct credentials. This will fail. • Verify the logs on TOE showing an account is locked out.
Expected Test Results	• The TOE should support user lockout after the configured number of unsuccessful login attempts and lock out time. • TOE should show account locked out logs.
Pass/Fail with Explanation	Pass. The TOE successfully locks out a user after a configured number of failed login attempts also once the authentication attempts limit is reached, authentication attempts with valid credentials are no longer successful. This lockout applies to both the SSH and GUI[29] methods by which remote administrators access the TOE (SSH and HTTPS). This meets the testing requirements.

7.2.3 FIA_AFL.1 TEST #2A [**TD0570**]

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests for each method by which remote administrators access the TOE (e.g. any passwords entered as part of establishing the connection protocol or the remote administrator application): Test 2: After reaching the limit for unsuccessful authentication attempts as in Test 1 above, the evaluator shall proceed as follows: If the **administrator action** selection in FIA_AFL.1.2 is included in the ST, then the evaluator shall confirm by testing that following the operational guidance and performing each action specified in the ST to re-enable the remote

[29] Only VX series models don't support Web UI.

	administrator's access results in successful access (when using valid credentials for that administrator).
Test Steps	SSH: • Attempt to connect to the TOE with incorrect credentials. • Verify after the final attempt that the user account is now locked out. • Manually unlock the user account. • Verify that the user account is unlocked. • Login with correct credentials. • Verify the lockout has been removed with logs. HTTPS: • Attempt to connect to the TOE with incorrect credentials. • Verify after the final attempt that the user account is now locked out. • Manually unlock the user account. • Verify that the user account is unlocked. • Login with correct credentials. • Verify the lockout has been removed with logs.
Expected Test Results	• The TOE should allow a locked-out user to log in again after the account is unlocked by the administrator. • TOE should show account locked out logs and successful authentication logs once account is unlocked by the administrator.
Pass/Fail with Explanation	Pass. For SSH and Web-UI the TOE successfully rejects login attempts with valid credentials for locked-out users and allows a locked-out user to log in again after their account is unlocked by an administrator. This meets the testing requirements.

7.2.4 FIA_AFL.1 TEST #2B [**TD0570**]

Item	**Data**
Test Assurance Activity	The evaluator shall perform the following tests for each method by which remote administrators access the TOE (e.g. any passwords entered as part of establishing the connection protocol or the remote administrator application): Test 2: After reaching the limit for unsuccessful authentication attempts as in Test 1 above, the evaluator shall proceed as follows: If the **time period** selection in FIA_AFL.1.2 is included in the ST, then the evaluator shall wait for just less than the time period configured in Test 1 and show that an authorisation attempt using valid credentials does not result in

	successful access. The evaluator shall then wait until just after the time period configured in Test 1 and show that an authorisation attempt using valid credentials results in successful access.
Test Steps	• Set user unlock time on the TOE. SSH: • Attempt to login with an incorrect password till the account lockout is triggered. • Verify the account is locked. • Verify that the account is locked via logs. • Attempt to login with the correct password and verify that it fails while an account is still locked. • Wait for lockout time to be over. Attempt to login with the correct password after lockout time is over, and verify it is successful. • Verify successful login with logs. HTTPS: • Attempt to login with an incorrect password till the account lockout is triggered. • Verify the account is locked. • Verify that the account is locked via logs. • Attempt to login with the correct password and verify that it fails while an account is still locked. • Attempt to login with the correct password after lockout time is over, and verify it is successful. • Verify successful login with logs.
Expected Test Results	• The TOE should allow a locked-out user to log in again after lockout time expires. • TOE should show account locked out logs and successful authentication logs once locked out time is completed.
Pass/Fail with Explanation	Pass. For SSH and Web-UI the TOE successfully rejects log in with valid credentials till the lockout period and allows a locked-out user to log in again after the lockout time expires. This meets the testing requirements.

7.2.5 FIA_PMG_EXT.1 TEST #1 [**TD0571**]

Item	Data
Test Assurance Activity	The evaluator shall compose passwords that meet the requirements in some way. For each password, the evaluator shall verify that the TOE supports the password. While the evaluator is not required (nor is it feasible) to test all possible compositions of passwords, the evaluator shall ensure that all characters, and a minimum length listed in the requirement are supported and justify the subset of those characters chosen for testing.

Test Steps	• Set the minimum password requirements.
	  ○ Minimum 15 character length
	  ○ Minimum 1 upper case
	  ○ Minimum 1 lower case
	  ○ Minimum 1 digit
	  ○ Minimum 1 special character
	
	SSH:
	
	• Attempt to create minimum 15 characters password with username: good & password: A'B1C+D7-E!a@bc1de
	• Attempt to create minimum 15 characters password with username: good1 & password: FG.2/HI:8J#f$gh2ij|
	• Attempt to create minimum 15 characters password with username: good2 & password: K;L3<MN9O%k^lm3no
	• Attempt to create minimum 15 characters password with username: good3 & password: P=Q>4RS{0}T&p*qr4st
	• Attempt to create minimum 15 characters password with username: good4 & password: U?V5W\X1Y(u)vw5xy
	• Attempt to create minimum 15 characters password with username: good5 & password: ZA6[B]C2`D!z@ab6cd
	• Attempt to create minimum 15 characters password with username: good6 & password: UV5W~X1Y_u)vw5xy
	• Verify all the usernames with correct password requirements are created.
	
	GUI:
	
	• Attempt to create minimum 15 characters password with username: correct & password: A'B1C+D7-E!a@bc1de
	• Attempt to create minimum 15 characters password with username: correct1 & password: FG.2/HI:8J#f$gh2ij|
	• Attempt to create minimum 15 characters password with username: correct2 & password: K;L3<MN9O%k^lm3no

	• Attempt to create minimum 15 characters password with username: correct3 & password: P=Q>4RS{0}T&p*qr4st • Attempt to create minimum 15 characters password with username: correct4 & password: U?V5W\X1Y(u)vw5xy • Attempt to create minimum 15 characters password with username: correct5 & password: ZA6[B]C2`D!z@ab6cd • Attempt to create minimum 15 characters password with username: correct6 & password: UV5W~X1Y_u)vw5xy • Verify all the usernames with correct password requirements are created.
Expected Test Results	• User accounts with passwords that meet requirements will be created. • Evidence – TOE logs showing successful creation of users.
Pass/Fail with Explanation	Pass. The TOE successfully creates user accounts with strong passwords. All characters claimed in the ST are supported by the TOE, and the passwords meet the minimum length requirement specified. This meets the testing requirements.

7.2.6 FIA_PMG_EXT.1 TEST #2 [**TD0571**]

Item	Data
Test Assurance Activity	The evaluator shall compose passwords that do not meet the requirements in some way. For each password, the evaluator shall verify that the TOE does not support the password. While the evaluator is not required (nor is it feasible) to test all possible compositions of passwords, the evaluator shall ensure that the TOE enforces the allowed characters and the minimum length listed in the requirement and justify the subset of those characters chosen for testing.
Test Steps	SSH: • Attempt to create a user with a missing upper case character in the password with username: bad & password: ab1cd7e!a@bc1de • Confirm that the user could not be created via logs. • Attempt to create a user with missing lowing case character in password with username: bad1 & password: FG2HI8J#F$GH2IJ • Confirm that the user could not be created via logs. • Attempt to create a user with missing digits in the password with username: bad2 & password: KLmMNra%k^lmsno • Confirm that the user could not be created via logs.

	• Attempt to create a user with a missing special character in the password with username: bad3 & password: PQ4RS0T2prqr4st • Confirm that the user could not be created via logs. • Attempt to create a user with less than 15 characters in password username: bad4 & password: UV5WX1Y(u)vw • Confirm that the user could not be created via logs. GUI: • Attempt to create a user with a missing upper case character in the password with username: incorrect & password: ab1cd7e!a@bc1de • Confirm that the user could not be created via logs. • Attempt to create a user with missing lowing case character in password with username: incorrect1 & password: FG2HI8J#F$GH2IJ • Confirm that the user could not be created via logs. • Attempt to create a user with missing digits in the password with username: incorrect2 & password: KLmMNra%k^lmsno • Confirm that the user could not be created via logs. • Attempt to create a user with a missing special character in the password with username: incorrect3 & password: PQ4RS0T2prqr4st • Confirm that the user could not be created via logs. • Attempt to create a user with less than 15 characters in password username: incorrect4 & password: UV5WX1Y(u)vw • Confirm that the user could not be created via logs.
Expected Test Results	• The TOE should generate an error when attempting to add users with incorrect password combinations, resulting in failure due to an 'Invalid Password' error. • Evidence - screenshot showing error while creating a user with an incorrect password. • TOE logs should show an 'Invalid Password' error.
Pass/Fail with Explanation	Pass. User accounts cannot be created without configured password requirements being met. This meets the testing requirements.

7.2.7 FIA_UIA_EXT.1 TEST #1

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests for each method by which administrators access the TOE (local and remote), as well as for each type of credential supported by the login method: Test 1: The evaluator shall use the guidance documentation to configure the appropriate credential supported for the login method. For that credential/login method, the evaluator shall show that providing correct I&A information results in the ability to access the system, while providing incorrect information results in denial of access.
Test Steps	GUI • Attempt to log into the device with incorrect credentials. Login will fail. • Verify the login attempt failure logs on TOE. • Attempt to log into the device with the correct credentials. This will succeed. • Verify the successful authentication logs on TOE. SSH • Attempt to log into the device with incorrect credentials. Login will fail. • Verify the login attempt failure logs on TOE. • Attempt to log into the device with the correct credentials. This will succeed. • Verify the successful authentication logs on TOE. Console • Attempt to log into the device with incorrect credentials. Login will fail. • Verify the login attempt failure logs on TOE. • Attempt to log into the device with the correct credentials. This will succeed. • Verify the successful authentication logs on TOE.
Expected Test Results	• The TOE should allow the user with correct credentials and reject the user with incorrect credentials. • TOE should generate logs for the successful and unsuccessful login attempts.
Pass/Fail with Explanation	Pass. Through GUI, SSH, and console the TOE successfully authenticates users with correct credentials and login fails when incorrect credentials are used. This meets the testing requirements.

7.2.8 FIA_UIA_EXT.1 TEST #2

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests for each method by which administrators access the TOE (local and remote), as well as for each type of credential supported by the login method: Test 2: The evaluator shall configure the services allowed (if any) according to the guidance documentation, and then determine the services available to an external remote entity. The evaluator shall determine that the list of services available is limited to those specified in the requirement.
Test Steps	Remote CLI: • Verify that before login, the only options presented are username/password prompt and banner. • The evaluator attempts to enter certain commands, such as show and config at the login screen. These commands fail. • Verify authentication logs reflect failure. Remote GUI: • Verify that before login, the only options presented are username/password prompt and banner. • Enter certain commands, this should fail. • Verify authentication logs reflect failure.
Expected Test Results	• The TOE should not expose services to an unauthenticated remote entity, and it should only display a banner. • Evidence – Snap showing only the username/password prompt and the banner is present before login.
Pass/Fail with Explanation	Pass. The TOE allows only a username/password prompt and the banner to be visible prior to login. This meets the testing requirements.

7.2.9 FIA_UIA_EXT.1 TEST #3

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests for each method by which administrators access the TOE (local and remote), as well as for each type of credential supported by the login method: Test 3: For local access, the evaluator shall determine what services are available to a local administrator prior to logging in, and make sure this list is consistent with the requirement.
Test Steps	• Connect to the TOE via the console and verify the only option presented is the username/password prompt and banner. • Attempt to execute authenticated commands such as show run, show logging, and configure terminal. This will fail. • Verify the TOE logs showing authentication failure.
Expected Test Results	• The TOE does not expose any services other than the ones meant to be exposed i.e. username/password prompt and banner. • Evidence – Snap shows only the username/password prompt and banner is present before login.
Pass/Fail with Explanation	Pass. No system services are available to a local administrator prior to logging in via the directly connected console. This meets the testing requirements.

7.2.10 FIA_UIA_EXT.1 TEST #4

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests for each method by which administrators access the TOE (local and remote), as well as for each type of credential supported by the login method: Test 4: For distributed TOEs where not all TOE components support the authentication of Security Administrators according to FIA_UIA_EXT.1 and FIA_UAU_EXT.2, the evaluator shall test that the components authenticate Security Administrators as described in the TSS.
Pass/Fail with Explanation	N/A. TOE is not distributed.

7.2.11 FIA_UAU.7 TEST #1

Item	Data
Test Assurance Activity	The evaluator shall perform the following test for each method of local login allowed: The evaluator shall locally authenticate to the TOE. While making this attempt, the evaluator shall verify that at most obscured feedback is provided while entering the authentication information.
Test Steps	• Log into the TOE via console. Verify that authentication information i.e., the password is obscured.
Expected Test Results	• The TOE should support the obscuring of passwords. • Evidence - screenshot showing password is obscured.
Pass/Fail with Explanation	Pass. TOE meets password obscurity standards. This meets the testing requirements.

7.2.12 FMT_MOF.1/MANUALUPDATE TEST #1

Item	Data
Test Assurance Activity	The evaluator shall try to perform the update using a legitimate update image without prior authentication as Security Administrator (either by authentication as a user with no administrator privileges or without user authentication at all – depending on the configuration of the TOE). The attempt to update the TOE shall fail.
Test Steps	• Login as a user without Security Administrator privileges. • Attempt to update the device and verify the command is rejected. • Verify the authentication logs generated on the TOE.
Expected Test Results	• TOE should not allow users without Security Administrator privileges to update using a legitimate update image. • Evidence - screenshot showing upgrade commands are rejected.

Pass/Fail with Explanation	Pass. The TOE does not allow users without Security Administrator privileges to update using a legitimate update image.This meets the testing requirements.

7.2.13 FMT_MOF.1/MANUALUPDATE TEST #2

Item	Data
Test Assurance Activity	The evaluator shall try to perform the update with prior authentication as Security Administrator using a legitimate update image. This attempt should be successful. This test case should be covered by the tests for FPT_TUD_EXT.1 already.
Pass/Fail with Explanation	Pass. This test has been completed as part of the requirements specified in FPT_TUD_EXT.1 Test#1.

7.2.14 FMT_MOF.1/FUNCTIONS (1) TEST #1

Item	Data
Test Assurance Activity	Test 1 (if **'transmission of audit data to external IT entity'** is selected from the second selection together with **'modify the behaviour of'** in the first selection): The evaluator shall try to modify all security related parameters for configuration of the transmission protocol for transmission of audit data to an external IT entity without prior authentication as Security Administrator (by authentication as a user with no administrator privileges or without user authentication at all). Attempts to modify parameters without prior authentication should fail. According to the implementation no other users than the Security Administrator might be defined and without any user authentication the user might not be able to get to the point where the attempt to modify the security related parameters can be executed. In that case it shall be demonstrated that access control mechanisms prevent execution up to the step that can be reached without authentication as Security Administrator.
Test Steps	• Login to the TOE as a user with no administrator privileges. • Attempt to modify TOE services and verify the command is rejected.

	• Verify the logs reflected for login.
Expected Test Results	• When an attempt to modify the audit data is made using an unprivileged user, it should fail. • The audit log should indicate that the user did not have prior authentication as a security administrator.
Pass/Fail with Explanation	Pass. Users without prior authentication/privilege as security administrators are unable to modify TOE services. This meets the testing requirements.

7.2.15 FMT_MOF.1/FUNCTIONS (1)TEST #2

Item	Data
Test Assurance Activity	Test 2 (if **'transmission of audit data to external IT entity'** is selected from the second selection together with **'modify the behaviour of'** in the first selection): The evaluator shall try to modify all security related parameters for configuration of the transmission protocol for transmission of audit data to an external IT entity with prior authentication as Security Administrator. The effects of the modifications should be confirmed. The evaluator does not have to test all possible values of the security related parameters for configuration of the transmission protocol for transmission of audit data to an external IT entity but at least one allowed value per parameter.
Test Steps	• Login to the TOE as a user with administrator privileges. • Attempt to modify TOE services and verify the command is accepted. • Verify TOE services are modified via logs.
Expected Test Results	• When an administrator tries to modify the audit data on the TOE, it should be successful. The command should be executed as the user has administrator privileges. • Audit log should show the user has prior authentication as a security administrator and that the user can modify the TOE services.
Pass/Fail with Explanation	Pass. Users with prior authentication/privilege as security administrators can modify TOE services. This meets the testing requirements.

7.2.16 FMT_MOF.1/FUNCTIONS (2) TEST #1

Item	Data
Test Assurance Activity	Test 1 (if **'handling of audit data'** is selected from the second selection together with **'modify the behaviour of'** in the first selection): The evaluator shall try to modify all security related parameters for configuration of the handling of audit data without prior authentication as Security Administrator (by authentication as a user with no administrator privileges or without user authentication at all). Attempts to modify parameters without prior authentication should fail. According to the implementation no other users than the Security Administrator might be defined and without any user authentication the user might not be able to get to the point where the attempt can be executed. In that case it shall be demonstrated that access control mechanisms prevent execution up to the step that can be reached without authentication as Security Administrator. The term 'handling of audit data' refers to the different options for selection and assignments in SFRs FAU_STG_EXT.1.2, FAU_STG_EXT.1.3 and FAU_STG_EXT.2/LocSpace.
Test Steps	• Login to the TOE as a user with no administrator privileges. • Attempt to modify the logging configuration and verify the command is rejected. • Verify the logs reflected for login.
Expected Test Results	• An unprivileged user should not be able to configure or make changes to the 'logging configuration' of the TOE. The command should not be executed as the user doesn't have the required privileges. • The audit log should indicate that the user did not have prior authentication as a security administrator.
Pass/Fail with Explanation	Pass. The users without prior authentication/privilege as security administrators could not modify the TOE audit data (logging configuration). This meets the testing requirements.

7.2.17 FMT_MOF.1/FUNCTIONS (2) TEST #2

Item	Data
Test Assurance Activity	Test 2 (if **'handling of audit data'** is selected from the second selection together with **'modify the behaviour of'** in the first selection): The evaluator shall try to modify all security related parameters for configuration of the handling of audit data with prior authentication as Security Administrator. The effects of the

	modifications should be confirmed. The term 'handling of audit data' refers to the different options for selection and assignments in SFRs FAU_STG_EXT.1.2, FAU_STG_EXT.1.3 and FAU_STG_EXT.2/LocSpace. The evaluator does not necessarily have to test all possible values of the security related parameters for configuration of the handling of audit data but at least one allowed value per parameter.
Test Steps	• Login to the TOE as a user with administrator privileges. • Attempt to modify TOE audit data (logging configuration) and verify that it is successful. • Verify TOE audit data (logging configuration) is modified via logs.
Expected Test Results	• A security administrator should be able to modify the 'logging configuration' of the TOE. The command should be executed successfully as the user is a security administrator. • Audit log should show a modification of TOE audit data (logging configuration).
Pass/Fail with Explanation	Pass. Only users with prior authentication/privilege as security administrators can modify TOE audit data (logging configuration). This meets the testing requirements.

7.2.18 FMT_MOF.1/FUNCTIONS (3) TEST #1

Item	Data
Test Assurance Activity	(if **'audit functionality when Local Audit Storage Space is full**' is selected from the second selection together with 'modify the behaviour of' in the first selection): The evaluator shall try to modify the behaviour when Local Audit Storage Space is full without prior authentication as Security Administrator (by authentication as a user with no administrator privileges or without user authentication at all). This attempt should fail. According to the implementation no other users than the Security Administrator might be defined and without any user authentication the user might not be able to get to the point where the attempt can be executed. In that case it shall be demonstrated that access control mechanisms prevent execution up to the step that can be reached without authentication as Security Administrator.
Pass/Fail with Explanation	N/A. The ST does not select 'audit functionality when Local Audit Storage Space is full'.

7.2.19 FMT_MOF.1/FUNCTIONS (3) TEST #2

Item	Data
Test Assurance Activity	(if **'audit functionality when Local Audit Storage Space is full**' is selected from the second selection together with 'modify the behaviour of' in the first selection): The evaluator shall try to modify the behaviour when Local Audit Storage Space is full with prior authentication as Security Administrator. This attempt should be successful. The effect of the change shall be verified. The evaluator does not necessarily have to test all possible values for the behaviour when Local Audit Storage Space is full but at least one change between allowed values for the behaviour
Pass/Fail with Explanation	N/A. The ST does not select 'audit functionality when Local Audit Storage Space is full'.

7.2.20 FMT_MOF.1/FUNCTIONS TEST #3

Item	Data
Test Assurance Activity	(if in the first selection **'determine the behaviour of'** has been chosen together with for any of the options in the second selection): The evaluator shall try to determine the behaviour of all options chosen from the second selection without prior authentication as Security Administrator (by authentication as a user with no administrator privileges or without user authentication at all). This can be done in one test or in separate tests. The attempt(s) to determine the behaviour of the selected functions without administrator authentication shall fail. According to the implementation no other users than the Security Administrator might be defined and without any user authentication the user might not be able to get to the point where the attempt can be executed. In that case it shall be demonstrated that access control mechanisms prevent execution up to the step that can be reached without authentication as Security Administrator.

Pass/Fail with Explanation	N/A. The ST does not select 'determine the behaviour of'.

7.2.21 FMT_MOF.1/FUNCTIONS TEST #4

Item	Data
Test Assurance Activity	(if in the first selection **'determine the behaviour of'** has been chosen together with for any of the options in the second selection): The evaluator shall try to determine the behaviour of all options chosen from the second selection with prior authentication as Security Administrator. This can be done in one test or in separate tests. The attempt(s) to determine the behaviour of the selected functions with Security Administrator authentication shall be successful.
Pass/Fail with Explanation	N/A. The ST does not select 'determine the behaviour of'.

7.2.22 FMT_MTD.1/COREDATA TEST #1

Item	Data
Test Assurance Activity	No separate testing for FMT_MTD.1/CoreData is required unless one of the management functions has not already been exercised under any other SFR.
Test Output	**This test is completed throughout the process of testing the following SFRs and there are no remaining functions to be tested:**

Functions	Test cases
Transmission of audit data to external IT entity	FMT_MOF.1/Functions (1)Test #2
Handling of audit data	FMT_MOF.1/Functions (2) Test #2

	Ability to manage the cryptographic keys	FMT_MTD.1/CryptoKeys Test #2
	Management functions	FMT_SMF.1 Test #1
Pass/Fail with Explanation	Pass. No separate testing for FMT_MTD.1/CoreData is required as all management functions have already been already exercised under claimed SFRs and there are no remaining functions to be tested.	

7.2.23 FMT_MTD.1/CRYPTOKEYS TEST #1

Item	Data
Test Assurance Activity	The evaluator shall try to perform at least one of the related actions (modify, delete, generate/import) without prior authentication as Security Administrator (either by authentication as a non-administrative user, if supported, or without authentication at all). Attempts to perform related actions without prior authentication should fail. According to the implementation no other users than the Security Administrator might be defined and without any user authentication the user might not be able to get to the point where the attempt to manage cryptographic keys can be executed. In that case it shall be demonstrated that access control mechanisms prevent execution up to the step that can be reached without authentication as Security Administrator.
Test Steps	• Login to the TOE as a user with no administrator privileges. • Attempt to modify cryptographic keys i.e. generate CSR and verify the command is not accepted. • Verify the logs reflected for login.
Expected Test Results	• When an attempt to modify the cryptographic keys is made using an unprivileged user, it should fail. • The audit log should indicate that the user did not have prior authentication as a security administrator.
Pass/Fail with Explanation	Pass. Users without prior authentication/privilege as security administrators cannot modify cryptographic keys. This meets the testing requirements.

7.2.24 FMT_MTD.1/CRYPTOKEYS TEST #2

Item	Data

Test Assurance Activity	The evaluator shall try to perform at least one of the related actions with prior authentication as Security Administrator. This attempt should be successful.
Test Steps	• Login to the TOE as a user with administrator privileges. • Attempt to modify cryptographic keys i.e. generate CSR and verify the command is accepted. • Verify CSR generation via TOE logs.
Expected Test Results	• When an attempt to modify the cryptographic keys is made using a privileged user, it should pass. • Audit log should show the user has prior authentication as a security administrator and that the user can modify the cryptographic keys i.e. generate CSR.
Pass/Fail with Explanation	Pass. Users with prior authentication/privilege as security administrators can modify cryptographic keys. This meets the testing requirements.

7.2.25 FMT_SMF.1 TEST #1

Item	**Data**
Test Assurance Activity	The evaluator tests management functions as part of testing the SFRs identified in section 2.4.4. No separate testing for FMT_SMF.1 is required unless one of the management functions in FMT_SMF.1.1 has not already been exercised under any other SFR.
Test Output	**This test is completed throughout the process of testing the following SFRs:**

Management Functions	**Test cases**
Ability to administer the TOE locally and remotely	FIA_UIA_EXT.1 Test #1
Ability to configure the access banner	FTA_TAB.1 Test #1

Ability to configure the session inactivity time before session termination or locking	FTA_SSL_EXT.1.1 Test #1, FTA_SSL.3 test #1
Ability to update the TOE, and to verify the updates using digital signature capability prior to installing those updates	FPT_TUD_EXT.1Test #1
Ability to configure the authentication failure parameters for FIA_AFL.1	FIA_AFL.1 Test #1 and FIA_AFL.1 Test#2b
Ability to configure audit behaviour (e.g. changes to storage locations for audit; changes to behaviour when local audit storage space is full);	FMT_MOF.1/Functions (2) Test #2 , FAU_STG_EXT.1
Ability to modify the behaviour of the transmission of audit data to an external IT entity	FMT_MOF.1/Functions(1)Test#2
Ability to manage the cryptographic keys	FMT_MTD.1/CryptoKeys Test #2
Ability to configure the cryptographic functionality	FCS_SSHS_EXT.1.5 Test #1
Ability to import X.509v3 certificates to the TOE's trust store	FIA_X509_EXT.1/ Rev Test #1a
Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors	FIA_X509_EXT.1/ Rev Test #1a

	Ability to set the time which is used for timestamps	FPT_STM_EXT.1 test #1
	Ability to re-enable an Administrator account	FIA_AFL.1 Test #2a
	Ability to configure NTP	FCS_NTP_EXT.1.1 Test #1 and FCS_NTP_EXT.1.2Test #1
	Ability to manage the trusted public keys database	FCS_SSHS_EXT.1.2 Test #1
Pass/Fail with Explanation	Pass. Throughout the various security functionality testing of the TOE, FMT_SMF.1 Specification of Management Functions requirements have been met. This meets the testing requirements.	

7.2.26 FMT_SMR.2 TEST #1

Item	Data
Test Assurance Activity	In the course of performing the testing activities for the evaluation, the evaluator shall use all supported interfaces, although it is not necessary to repeat each test involving an administrative action with each interface. The evaluator shall ensure, however, that each supported method of administering the TOE that conforms to the requirements of this cPP be tested; for instance, if the TOE can be administered through a local hardware interface; SSH; and TLS/HTTPS; then all three methods of administration must be exercised during the evaluation team's test activities.
Pass/Fail with Explanation	Pass. There are three interfaces where these can be tested (console/Remote CLI/Remote GUI) and all test cases use these interfaces. The evaluator has met this requirement through the execution of the entirety of this test report by performing actions via all three interfaces.

7.2.27 FTA_SSL.3 TEST #1

Item	Data
Test Assurance Activity	The evaluator follows the guidance documentation to configure several different values for the inactivity time period referenced in the component. For each period configured, the evaluator establishes a remote interactive session with the TOE. The evaluator then observes that the session is terminated after the configured time period.
Test Steps	Remote CLI: • Configure the TOE with a maximum inactivity time period of one minute. • Log into the TOE via remote connection. • Allow the session to time out. • Verify the logs for session timeout. • Configure the TOE with a maximum inactivity time period of two minutes. • Log into the TOE via remote connection. • Allow the session to time out. • Verify the logs for session timeout. • Configure the TOE with a maximum inactivity time period of five minutes. • Log into the TOE via remote connection. • Allow the session to time out. • Verify the logs for session timeout. Remote GUI: • Configure the TOE with a maximum inactivity time period of one minute. • Log into the TOE via remote connection. • Allow the session to time out. • Verify the logs for session timeout. • Configure the TOE with a maximum inactivity time period of two minutes. • Log into the TOE via remote connection. • Allow the session to time out. • Verify the logs for session timeout. • Configure the TOE with a maximum inactivity time period of five minutes. • Log into the TOE via remote connection. • Allow the session to time out. • Verify the logs for session timeout.
Expected Test Results	• The TOE should support configuration for several different values for the inactivity time period and successfully terminate the session after the timeout period. • TOE should generate logs for session timeout.

Pass/Fail with Explanation	Pass. The TOE disconnects users from remote interactive sessions after meeting the inactivity time limit. This meets the testing requirements.

7.2.28 FTA_SSL.4 TEST #1

Item	Data
Test Assurance Activity	The evaluator initiates an interactive local session with the TOE. The evaluator then follows the guidance documentation to exit or log off the session and observes that the session has been terminated.
Test Steps	• Log in to TOE via a local console connection. • Log off from TOE. • Verify the logs for user logout.
Expected Test Results	• The TOE should terminate the local session after the user logs off. • TOE should generate logs for session timeout.
Pass/Fail with Explanation	Pass. The TOE allows the user to terminate the directly connected administrative session. This meets the testing requirements.

7.2.29 FTA_SSL.4 TEST #2

Item	Data
Test Assurance Activity	The evaluator initiates an interactive remote session with the TOE. The evaluator then follows the guidance documentation to exit or log off the session and observes that the session has been terminated.
Test Steps	Remote CLI: • Log into the TOE via remote session. • Log out of the device. • Verify via logs. Remote GUI:

	• Log into the TOE via remote session. • Log out of the device. • Verify via logs.
Expected Test Results	• The TOE should terminate the remote session after the user logs off. • TOE should generate logs for logout.
Pass/Fail with Explanation	Pass. The TOE allows the user to terminate the remote interactive session. This meets the testing requirements.

7.2.30 FTA_SSL_EXT.1.1 TEST #1

Item	Data
Test Assurance Activity	The evaluator follows the guidance documentation to configure several different values for the inactivity time period referenced in the component. For each period configured, the evaluator establishes a local interactive session with the TOE. The evaluator then observes that the session is either locked or terminated after the configured time period. If locking was selected from the component, the evaluator then ensures that reauthentication is needed when trying to unlock the session.
Test Steps	• Configure the TOE with a maximum inactivity time period of one minute. • Log into the TOE via the local console. • Allow the session to time out. • Verify the logs for session timeout. • Configure the TOE with a maximum inactivity time period of two minutes. • Log into the TOE via the local console. • Allow the session to time out. • Verify the logs for session timeout. • Configure the TOE with a maximum inactivity time period of five minutes. • Log into the TOE via the local console. • Allow the session to time out. • Verify the logs for session timeout.
Expected Test Results	• The TOE should terminate the session after the configured time period. • TOE logs should show session termination.

Pass/Fail with Explanation	Pass. TOE terminates the user session on the local console after the inactivity time limit is reached. This meets the testing requirements.

7.2.31 FTA_TAB.1 TEST #1

Item	Data
Test Assurance Activity	The evaluator follows the guidance documentation to configure a notice and consent warning message. The evaluator shall then, for each method of access specified in the TSS, establish a session with the TOE. The evaluator shall verify that the notice and consent warning message is displayed in each instance.
Test Steps	• Configure banner for the SSH and GUI. • Configure banner for the Console. Console: • login and verify that the banner is being displayed. SSH: • login and verify that the banner is being displayed. GUI • login and verify that the banner is being displayed.
Expected Test Results	• The TOE should support the display of banners. • Evidence - screenshot showing banners.
Pass/Fail with Explanation	Pass. An access banner can be set for all the methods (Console, SSH and GUI) that can be used to access the device. This meets the testing requirements.

7.2.32 FTP_TRP.1/ADMIN TEST #1

Item	Data
Test Assurance Activity	The evaluators shall ensure that communications using each specified (in the guidance documentation) remote administration method is tested during the course of the evaluation, setting up the connections as described in the guidance documentation and ensuring that communication is successful.
Test Steps	HTTPS: • Log into the TOE via HTTPS. • Verify the audit logs to confirm that the user successfully logs in to the TOE. • Verify that the session was established, and data is encrypted via packet capture. SSH: • Log into the TOE via SSH. • Verify the audit logs to confirm that the user successfully logs in to the TOE. • Verify that the session was established, and data is encrypted via packet capture.
Expected Test Results	• The TOE should encrypt the traffic successfully. • Evidence – Packet capture showing successful connection. • TOE logs should show a successful login.
Pass/Fail with Explanation	Pass. Users are successfully able to access the TOE via TLS and SSH connection. This meets the testing requirements.

7.2.33 FTP_TRP.1/ADMIN TEST #2

Item	Data
Test Assurance Activity	The evaluator shall ensure, for each communication channel, the channel data is not sent in plaintext.
Pass/Fail with Explanation	Pass. Refer to FTP_TRP.1/Admin Test #1 for encrypted channel data.

7.3 SSHS

7.3.1 FCS_SSHS_EXT.1.2 TEST #1 [**TD0631**]

Item	Data
Test Assurance Activity	Test objective: The purpose of these tests is to verify server supports each claimed client authentication method. Test 1: For each supported client public-key authentication algorithm, the evaluator shall configure a remote client to present a public key corresponding to that authentication method (e.g., 2048-bit RSA key when using ssh-rsa public key). The evaluator shall establish sufficient separate SSH connections with an appropriately configured remote non-TOE SSH client to demonstrate the use of all applicable public key algorithms. It is sufficient to observe the successful completion of the SSH Authentication Protocol to satisfy the intent of this test.
Test Steps	• Generate the ssh-rsa pub key (Key Size 3072). • Configure the TOE to support the RSA-based SSH authentication method. • Log into the TOE via SSH with RSA-based authentication. • Verify the successful authentication via packet capture. • Verify the successful authentication via logs. • Generate the rsa-sha2-512 pub key (Key Size 2048). • Configure the TOE to support the RSA-based SSH authentication method. • Log into the TOE via SSH with RSA-based authentication. • Verify the successful authentication via packet capture. • Verify the successful authentication via logs. • Generate the rsa-sha2-256 pub key (Key Size 2048). • Configure the TOE to support the RSA-based SSH authentication method. • Log into the TOE via SSH with RSA-based authentication. • Verify the successful authentication via packet capture. • Verify the successful authentication via logs.

Expected Test Results	• The TOE should successfully establish the SSH session connection with the client using public key authentication. • Log should show the successful connection of each algorithm. • Packet capture showing the successful connection for each algorithm.
Pass/Fail with Explanation	Pass. The TOE successfully establishes the SSH session with the client using the supported public key algorithms. This meets the testing requirements.

7.3.2 FCS_SSHS_EXT.1.2 TEST #2 [**TD0631**]

Item	**Data**
Test Assurance Activity	Test 2: The evaluator shall choose one client public key authentication algorithm supported by the TOE. The evaluator shall generate a new client key pair for that supported algorithm without configuring the TOE to recognize the associated public key for authentication. The evaluator shall use an SSH client to attempt to connect to the TOE with the new key pair and demonstrate that authentication fails.
Test Steps	• Configure the SSH client with a new RSA keypair for SSH without configuring the TOE. • Log into the TOE SSH using RSA-based authentication. • Verify authentication failed using public key via logs. • Verify the packet capture showing SSH connection.
Expected Test Results	• The TOE should reject SSH connections when incorrect/unknown public keys are presented. • Evidence: Screenshot/CLI output should show that authentication via public key has failed, and a password prompt is generated. • TOE logs should show authentication failure using the public key.
Pass/Fail with Explanation	Pass. The TOE does not allow public key authentication if the public key of the SSH user has not been uploaded to the TOE. This meets the testing requirements.

7.3.3 FCS_SSHS_EXT.1.2 TEST #3 [**TD0631**]

Item	**Data**

Test Assurance Activity	Test 3: [Conditional] If Password-based authentication method has been selected in the FCS_SSHS_EXT.1.2, the evaluator shall configure the TOE to accept Password-based authentication and demonstrate that user authentication succeeds when the correct Password is provided by the connecting SSH client.
Test Steps	• Ensure the TOE supports Password-based authentication. • Log into the TOE via SSH with Password authentication. • Verify the successful authentication logs. • Verify via packet capture that the SSH session was established.
Expected Test Results	• The TOE should set up a user with Password-based authentication and user authentication succeeds when the correct Password is provided by the user. • Packet capture should show the SSH session being established. • Log should show successful authentication using password-based authentication.
Pass/Fail with Explanation	Pass. The TOE successfully accepts Password-based authentication from a remote SSH client. This meets the testing requirements.

7.3.4 FCS_SSHS_EXT.1.2 TEST #4 [**TD0631**]

Item	**Data**
Test Assurance Activity	Test 4: [Conditional] If Password-based authentication method has been selected in the FCS_SSHS_EXT.1.2, the evaluator shall configure the TOE to accept Password-based authentication and demonstrate that user authentication fails when the incorrect Password is provided by the connecting SSH client.
Test Steps	• Ensure the TOE supports Password-based authentication. • Attempt to Log into the TOE via SSH with the correct username and incorrect Password-based authentication parameters (connection will fail). • Verify authentication failure via logs. • Verify authentication failure via packet capture.
Expected Test Results	• The TOE should set up a user with Password-based authentication. • User authentication should fail when an incorrect Password is provided by the user. • Packet capture should show the connection is closed. • Log should show unsuccessful authentication.

Pass/Fail with Explanation	Pass. The TOE does not establish a connection with a remote SSH user when incorrect authentication credentials are presented. This meets the testing requirements.

7.3.5 FCS_SSHS_EXT.1.3 TEST #1

Item	Data
Test Assurance Activity	The evaluator shall demonstrate that if the TOE receives a packet larger than that specified in this component, that packet is dropped.
Test Steps	• Establish an SSH connection to the TOE via the 'acumen-sshs' Tool and send a packet larger than the established limit. • Verify the error logs generated on the TOE due to a large packet. • Verify via packet capture that the large packet is dropped.
Expected Test Results	• The TOE should drop a packet larger than the allowed range. • The TOE should generate error logs when the sent packet exceeds the allowed range. • Packet capture should show TOE closes the connection when the sent packet exceeds the allowed range.
Pass/Fail with Explanation	Pass. The TOE drops large packets that are received within an SSH session. This meets the testing requirements.

7.3.6 FCS_SSHS_EXT.1.4 TEST #1

Item	Data
Test Assurance Activity	The evaluator must ensure that only claimed ciphers and cryptographic primitives are used to establish an SSH connection. To verify this, the evaluator shall start session establishment for an SSH connection from a remote client (referred to as 'remote endpoint' below). The evaluator shall capture the traffic exchanged between the TOE and the remote endpoint during protocol negotiation (e.g. using a packet capture tool or information provided by the endpoint, respectively). The evaluator shall verify from the captured traffic that the TOE offers all the ciphers defined in the TSS for

	the TOE for SSH sessions, but no additional ones compared to the definition in the TSS. The evaluator shall perform one successful negotiation of an SSH session to verify that the TOE behaves as expected. It is sufficient to observe the successful negotiation of the session to satisfy the intent of the test. If the evaluator detects that not all ciphers defined in the TSS for SSH are supported by the TOE and/or the TOE supports one or more additional ciphers not defined in the TSS for SSH, the test shall be regarded as failed.
Test Steps	• Configure the TOE to support SSH. • Establish an SSH session with the TOE. • Verify the ciphers offered by the TOE via packet capture. • Verify with logs that a session was established.
Expected Test Results	• The TOE should establish the SSH session only with the claimed encryption algorithms. • Packet capture should show that TOE only offers ciphers: aes128-ctr, aes256-ctr, aes128-gcm@openssh.com, aes256-gcm@openssh.com. • TOE log should show successful negotiation of the SSH session when claimed encryption algorithms are used.
Pass/Fail with Explanation	Pass. The TOE can successfully establish the SSH session with the client using only the claimed encryption algorithms. This meets the testing requirements.

7.3.7 FCS_SSHS_EXT.1.5 TEST #1 [**TD0631**]

Item	**Data**
Test Assurance Activity	Test objective: This test case is meant to validate that the TOE server will support host public keys of the claimed algorithm types. Test 1: The evaluator shall configure (only if required by the TOE) the TOE to use each of the claimed host public key algorithms. The evaluator will then use an SSH client to confirm that the client can authenticate the TOE server public key using the claimed algorithm. It is sufficient to observe (on the wire) the successful negotiation of the algorithm to satisfy the intent of the test.
Test Steps	• Generate a host-key (2048 bit) on the TOE. • Verify the logs generated for host key generation.

	• Established a session with the TOE using the rsa-sha2-256 host key algorithms. • Verify through logs that the connection is established successfully. • Verify via packet capture that the configured host key algorithm was used. • Established a session with the TOE using the rsa-sha2-512 host key algorithms. • Verify through logs that the connection is established successfully. • Verify via packet capture that the configured host key algorithm was used. • Generate a host-key (3072 bit) on the TOE. • Verify the logs generated for host key generation. • Established a session with the TOE using the ssh-rsa host key algorithms. • Verify through logs that the connection is established successfully. Verify via packet capture that the configured host key algorithm was used.
Expected Test Results	• TOE should establish a successful SSH connection only with the claimed host key algorithms. • TOE logs should show that the connection is established successfully. • Packet capture should show that the configured host key algorithm was used for the connection.
Pass/Fail with Explanation	Pass. The TOE establishes a successful SSH connection using each one of the claimed host public key algorithms. This meets the testing requirements.

7.3.8 FCS_SSHS_EXT.1.5 TEST #2 [**TD0631**]

Item	**Data**
Test Assurance Activity	Test objective: This negative test case is meant to validate that the TOE server does not support host public key algorithms that are not claimed. Test 2: The evaluator shall configure a non-TOE SSH client to only allow it to authenticate an SSH server host public key algorithm that is not included in the ST selection. The evaluator shall attempt to establish an SSH connection from the

	non-TOE SSH client to the TOE SSH server and observe that the connection is rejected.
Test Steps	• Established a session with the client using the unsupported host key algorithms (SSH-DSS). • Verify through logs that the SSH session was not established. • Verify through packet capture that the SSH session was not established.
Expected Test Results	• TOE should reject a connection request from an unclaimed host public key algorithm. • Packet capture should show failure due to a non-supported host key algorithm. • TOE logs should show the SSH session was not established.
Pass/Fail with Explanation	Pass. TOE rejects the connection when an unsupported host key algorithm is offered by the client while establishing the connection. This meets the testing requirement.

7.3.9 FCS_SSHS_EXT.1.6 TEST #1

Item	Data
Test Assurance Activity	Test 1: [conditional, if an **HMAC or AEAD_AES_*_GCM** algorithm is selected in the ST] The evaluator shall establish an SSH connection using each of the algorithms, except "implicit", specified by the requirement. It is sufficient to observe (on the wire) the successful negotiation of the algorithm to satisfy the intent of the test. Note: To ensure the observed algorithm is used, the evaluator shall ensure a non-aes*-gcm@openssh.com encryption algorithm is negotiated while performing this test.
Test Steps	• Establish an SSH session with the configured supported algorithms (HMAC-SHA2-256). • Verify that the SSH session was established using HMAC-SHA2-256 via packet capture. • Verify that the SSH session was established using HMAC-SHA2-256 via log. • Establish an SSH session with the configured supported algorithms (HMAC-SHA2-512). • Verify that the SSH session was established using HMAC-SHA2-512 via packet capture.

	• Verify that the SSH session was established using HMAC-SHA2-512 via log.
Expected Test Results	• The TOE should establish SSH connections with each claimed HMAC algorithm. • Packet capture should show the connection is established using the configured HMAC algorithm. • TOE logs should show the connection is established using the configured HMAC algorithm.
Pass/Fail with Explanation	Pass. The TOE can successfully establish SSH connections with each claimed data integrity algorithm. This meets the testing requirements.

7.3.10 FCS_SSHS_EXT.1.6 TEST #2

Item	Data
Test Assurance Activity	Test 2: [conditional, if an **HMAC or AEAD_AES_*_GCM** algorithm is selected in the ST] The evaluator shall configure an SSH client to only allow a MAC algorithm that is not included in the ST selection. The evaluator shall attempt to connect from the SSH client to the TOE and observe that the attempt fails. Note: To ensure the proposed MAC algorithm is used, the evaluator shall ensure a non-aes*-gcm@openssh.com encryption algorithm is negotiated while performing this test.
Test Steps	• Attempt to establish an SSH session using HMAC-MD5-96. • Verify via packet capture that the TOE rejects the connection. • Verify failure logs on the TOE.
Expected Test Results	• The TOE should reject SSH connections using an unsupported HMAC algorithm for data integrity. • Packet capture should show connection failure when an unsupported HMAC algorithm is used. • TOE logs should show the error message when an unsupported HMAC algorithm is used.
Pass/Fail with Explanation	Pass. The TOE rejects the connection when an unsupported MAC algorithm is offered while establishing an SSH session. This meets the testing requirements.

7.3.11 FCS_SSHS_EXT.1.7 TEST #1

Item	Data
Test Assurance Activity	The evaluator shall configure an SSH client to only allow the diffie-hellman-group1-sha1 key exchange. The evaluator shall attempt to connect from the SSH client to the TOE and observe that the attempt fails.
Test Steps	• Attempt to establish a connection with the TOE from an SSH client using Diffie-hellman-group1-sha1 as the key exchange method. • Verify connection failure via packet capture. • Verify that the session was not established via TOE logs.
Expected Test Results	• The TOE should permit connections when using Diffie-Hellman-group1-sha1. • The packet capture should show the TOE closing the connection when the kex_algorithm from the SSH client is unsupported. • TOE logs should show connection failure when the kex_algorithm from the SSH client is unsupported.
Pass/Fail with Explanation	Pass. The TOE rejects the connection when an unsupported algorithm (Diffie-hellman-group1-sha1) is used in the key exchange while establishing an SSH connection. This meets the testing requirements.

7.3.12 FCS_SSHS_EXT.1.7 TEST #2

Item	Data
Test Assurance Activity	For each allowed key exchange method, the evaluator shall configure an SSH client to only allow that method for key exchange, attempt to connect from the client to the TOE, and observe that the attempt succeeds.
Test Steps	• Attempt to establish a connection with the TOE from an SSH client using diffie-hellman-group16-sha512 as the key exchange method. • Verify that the session was established via logs. • Verify that the session was established via packet capture. • Attempt to establish a connection with the TOE from an SSH client using diffie-hellman-group18-sha512 as the key exchange method. • Verify that the session was established via logs. • Verify that the session was established via packet capture. • Attempt to establish a connection with the TOE from an SSH client using diffie-hellman-group14-sha256 as the key exchange method.

	• Verify that the session was established via logs. • Verify that the session was established via packet capture. • Attempt to establish a connection with the TOE from an SSH client using diffie-hellman-group14-sha1 as the key exchange method. • Verify that the session was established via logs. • Verify that the session was established via packet capture.
Expected Test Results	• The TOE should successfully establish SSH connections using diffie-hellman-group16-sha512, diffie-hellman-group18-sha512, diffie-hellman-group14-sha256, diffie-hellman-group14-sha1. • Packet capture should show a successful connection when a supported key exchange algorithm is used. • TOE logs should show a successful connection when a supported key exchange algorithm is used.
Pass/Fail with Explanation	Pass. The TOE can successfully establish SSH connections with each claimed key exchange method. This meets the testing requirements.

7.3.13 FCS_SSHS_EXT.1.8 TEST #1A

Item	Data
Test Assurance Activity	The evaluator needs to perform testing that rekeying is performed according to the description in the TSS. The evaluator shall test both, the **time-based threshold** and the traffic-based threshold. For testing of the time-based threshold, the evaluator shall use an SSH client to connect to the TOE and keep the session open until the threshold is reached. The evaluator shall verify that the SSH session has been active longer than the threshold value and shall verify that the TOE initiated a rekey (the method of verification shall be reported by the evaluator). Testing does not necessarily have to be performed with the threshold configured at the maximum allowed value of one hour of session time but the value used for testing shall not exceed one hour. The evaluator needs to ensure that the rekeying has been initiated by the TOE and not by the SSH client that is connected to the TOE. If one or more thresholds that are checked by the TOE to fulfil the SFR are configurable, the evaluator needs to verify that the threshold(s) can be configured as described in the guidance documentation and the evaluator needs to test that

	modification of the thresholds is restricted to Security Administrators (as required by FMT_MOF.1/Functions).
Test Steps	• Verify the time-based threshold for TOE. • Initiate a new SSH session using the 'acumen-sshs' tool and send traffic till the time-based threshold is met. • Verify via logs that rekey takes place after the time-based threshold.
Expected Test Results	• The TOE should issue a rekey after the specified time as configured on the TOE. • TOE logs should show the Session rekey request has been sent after a time-based threshold has been reached.
Pass/Fail with Explanation	Pass. The TOE initiates a rekey after the time-based threshold. This meets the testing requirements.

7.3.14 FCS_SSHS_EXT.1.8 TEST #1B

Item	Data
Test Assurance Activity	The evaluator needs to perform testing that rekeying is performed according to the description in the TSS. The evaluator shall test both, the time-based threshold and the **traffic-based** threshold. For testing of the traffic-based threshold the evaluator shall use the TOE to connect to an SSH client and shall transmit data to and/or receive data from the TOE within the active SSH session until the threshold for data protected by either encryption key is reached. It is acceptable if the rekey occurs before the threshold is reached (e.g. because the traffic is counted according to one of the alternatives given in the Application Note for FCS_SSHS_EXT.1.8). The evaluator shall verify that more data has been transmitted within the SSH session than the threshold allows and shall verify that the TOE initiated a rekey (the method of verification shall be reported by the evaluator). Testing does not necessarily have to be performed with the threshold configured at the maximum allowed value of one gigabyte of transferred traffic but the value used for testing shall not exceed one gigabyte. The evaluator needs to ensure that the rekeying has been initiated by the TOE and not by the SSH client that is connected to the TOE. If one or more thresholds that are checked by the TOE to fulfil the SFR are configurable, the evaluator needs to verify that the threshold(s) can be configured as described in the guidance documentation and the evaluator needs to test that

	modification of the thresholds is restricted to Security Administrators (as required by FMT_MOF.1/Functions). In cases where data transfer threshold could not be reached due to hardware limitations it is acceptable to omit testing of this (SSH rekeying based on data transfer threshold) threshold if both the following conditions are met: a) An argument is present in the TSS section describing this hardware- based limitation and b) All hardware components that are the basis of such argument are definitively identified in the ST. For example, if specific Ethernet Controller or WiFi radio chip is the root cause of such limitation, these chips must be identified.
Test Steps	• Verify the traffic-based threshold for TOE. • Initiate a new SSH session using 'acumen-sshs' tool and start sending traffic. • Verify via logs that rekey takes place after reaching the traffic-based threshold.
Expected Test Results	• The TOE should issue a rekey after the specified amount of data is transferred as configured on the TOE. • TOE logs should show session rekey requests being sent after reaching the set data limit.
Pass/Fail with Explanation	Pass. The TOE issues a rekey after the specified amount of data is sent. This meets the testing requirement.

7.4 TLSC

7.4.1 FCS_TLSC_EXT.1.1 TEST #1

Item	Data
Test Assurance Activity	The evaluator shall establish a TLS connection using each of the ciphersuites specified by the requirement. This connection may be established as part of the establishment of a higher-level protocol, e.g., as part of an HTTPS session. It is sufficient to observe the successful negotiation of a ciphersuite to satisfy the intent of the test; it is not necessary to examine the characteristics of the encrypted traffic in an attempt to discern the ciphersuite being used (for example, that the cryptographic algorithm is 128-bit AES and not 256-bit AES).

Test Steps	• Configure the TOE to connect to the TLS server. • Establish a connection with the TOE over TLS using the ciphersuite TLS_DHE_RSA_WITH_AES_128_CBC_SHA. • Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_DHE_RSA_WITH_AES_256_CBC_SHA. • Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256. • Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256. • Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256. • Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384. • Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256. • Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384. • Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256. • Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384. • Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256.

	• Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384. • Verify the required ciphersuite with packet capture.
Expected Test Results	• TOE should successfully establish a TLS connection with claimed ciphersuites. • Packet Captures should show the successful establishment of TLS connection with configured ciphersuites.
Pass/Fail with Explanation	Pass. TOE successfully negotiates each of the claimed cipher suites. This meets the test requirements.

7.4.2 FCS_TLSC_EXT.1.1 TEST #2

Item	Data
Test Assurance Activity	The evaluator shall attempt to establish the connection using a server with a server certificate that contains the Server Authentication purpose in the extendedKeyUsage field and verify that a connection is established. The evaluator will then verify that the client rejects an otherwise valid server certificate that lacks the Server Authentication purpose in the extendedKeyUsage field, and a connection is not established. Ideally, the two certificates should be identical except for the extendedKeyUsage field.
Test Steps	Valid Certificate: • Load the server certificate containing the Server Authentication purpose on the TLS server. • Establish a connection with the TOE over TLS and verify that it is successful. • Verify the successful connection with packet capture. Invalid Certificate: • Load the server certificate lacking the Server Authentication purpose on the TLS server. • Establish a connection with the TOE over TLS and verify that it is unsuccessful. • Verify the error logs on the device showing the connection is rejected due to an unsupported certificate purpose. • Verify the unsuccessful connection with packet capture.

Expected Test Results	• TOE should establish a connection with a server with an authorized server certificate, packet capture shows a successful connection. • TOE should reject the connection when a certificate lacking the Server Authentication purpose in the extendedKeyUsage filed is used, packet capture, and TOE logs show the connection failure due to invalid certificate extensions.
Pass/Fail with Explanation	Pass. TOE successfully established the connection using a server with a server certificate that contains the Server Authentication purpose in the extendedKeyUsage field and TOE rejects an otherwise valid server certificate that lacks the Server Authentication purpose in the extendedKeyUsage field. This meets the test requirements.

7.4.3 FCS_TLSC_EXT.1.1 TEST #3

Item	Data
Test Assurance Activity	The evaluator shall send a server certificate in the TLS connection that the does not match the server-selected ciphersuite (for example, send a ECDSA certificate while using the TLS_RSA_WITH_AES_128_CBC_SHA ciphersuite). The evaluator shall verify that the TOE disconnects after receiving the server's Certificate handshake message.
Test Steps	• Start the server using the 'acumen-tlsc-v2.2e' tool with a certificate that does not match the server-selected ciphersuite (an RSA certificate and ECDSA cipher suite) and verify that it fails. • Verify the error logs on the device showing the wrong certificate type. • Verify the unsuccessful connection with packet capture.
Expected Test Results	• The TOE should reject the connection with the server certificate that does not match the server-selected cipher suite. • The TOE logs and packet capture should indicate a connection failure when a server certificate that does not match the server-selected cipher suite is presented.
Pass/Fail with Explanation	Pass. The TOE denied a connection to a server using a certificate that doesn't match the ciphersuite. This meets the test requirements.

7.4.4 FCS_TLSC_EXT.1.1 TEST #4A

Item	Data
Test Assurance Activity	The evaluator shall configure the server to select the TLS_NULL_WITH_NULL_NULL ciphersuite and verify that the client denies the connection.
Test Steps	• Start the server using the 'acumen-tlsc-v2.2e' and send a server hello selecting TLS_NULL_WITH_NULL NULL cipher suite and verify the output. • Verify the error logs on the device showing failure due to an unknown cipher. • Verify the unsuccessful connection with packet capture.
Expected Test Results	• The TOE should reject a connection when the server selects a non-supported algorithm. • TOE logs should show connection failure due to an unknown cipher. • Packet capture should show that the TOE generates a fatal error when the server presents a null cipher suite.
Pass/Fail with Explanation	Pass. The TOE denies the session because TLS_NULL_WITH_NULL_NULL is presented. This meets the test requirements.

7.4.5 FCS_TLSC_EXT.1.1 TEST #4B

Item	Data
Test Assurance Activity	Modify the server's selected ciphersuite in the Server Hello handshake message to be a ciphersuite not presented in the Client Hello handshake message. The evaluator shall verify that the client rejects the connection after receiving the Server Hello.
Test Steps	• Start the server using the 'acumen-tlsc-v2.2e' tool and verify the connection with an unsupported ciphersuite. • Verify the error logs on the device showing connection failure due to the wrong cipher. • Verify the unsuccessful connection with packet capture.
Expected Test Results	• Client should reject the connection when the server modifies a ciphersuite. • TOE logs should show connection failure due to the wrong cipher. • Packet capture should show a fatal error generated by TOE after receiving the server hello as the wrong cipher is presented by server.

Pass/Fail with Explanation	Pass. The TOE rejects the connection with the wrong cipher by sending a Fatal Alert. This meets the testing requirements.

7.4.6 FCS_TLSC_EXT.1.1 TEST #4C

Item	Data
Test Assurance Activity	[conditional]: If the TOE presents the **Supported Elliptic Curves/Supported Groups Extension** the evaluator shall configure the server to perform an ECDHE or DHE key exchange in the TLS connection using a non-supported curve/group (for example P-192) and shall verify that the TOE disconnects after receiving the server's Key Exchange handshake message.
Test Steps	• Start the server using the 'acumen-tlsc-v2.2e' tool and verify the connection with an unsupported elliptical curve. • Verify the error logs on the device showing connection due to the wrong curve. • Verify the unsuccessful connection with packet capture.
Expected Test Results	• TOE should reject the connection if an unsupported curve is provided. • TOE logs should show connection failure due to the wrong curve. • Packet capture should show fatal error is generated by TOE after receiving the server's key exchange handshake message.
Pass/Fail with Explanation	Pass. When configured the server to perform an ECDHE key exchange in the TLS connection using a non-supported curve the connection fails. This meets the requirements.

7.4.7 FCS_TLSC_EXT.1.1 TEST #5A

Item	Data
Test Assurance Activity	Change the TLS version selected by the server in the Server Hello to a non-supported TLS version and verify that the client rejects the connection.
Test Steps	• Start the server using the 'acumen-tlsc-v2.2e' tool and send a server hello using an unsupported TLS version and verify that the TOE rejects the connection. • Verify connection failure logs due to wrong version number.

	• Verify the connection fails with packet capture.
Expected Test Results	• TOE should reject the connection when the server sends a message with a non-supported TLS version. • TOE logs should show connection failure due to an unsupported TLS version. • Packet capture should show a fatal error is generated by TOE as server hello using an unsupported TLS version is sent.
Pass/Fail with Explanation	Pass. When the TLS version selected by the server in the Server Hello is changed to a non-supported TLS version then the TOE rejects the connection. This meets the test requirements.

7.4.8 FCS_TLSC_EXT.1.1 TEST #5B

Item	Data
Test Assurance Activity	[conditional]: If **using DHE or ECDH**, modify the signature block in the Server's Key Exchange handshake message, and verify that the handshake does not finished successfully, and no application data flows. This test does not apply to cipher suites using RSA key exchange. If a TOE only supports RSA key exchange in conjunction with TLS, then this test shall be omitted.
Test Steps	• Start the server using the 'acumen-tlsc-v2.2e' tool and verify the connection when a signature byte is modified in the Server's Key Exchange handshake message. • Verify the error logs on the device showing connection failure due to a bad signature. • Verify the unsuccessful connection with packet capture.
Expected Test Results	• The connection establishment should fail when a signature byte is modified in the server's key exchange handshake message. • TOE logs should show connection failure due to a bad signature. • Packet capture should show fatal error is generated by TOE and the handshake does not finish successfully.
Pass/Fail with Explanation	Pass. The TOE rejects the connection due to the modified block in the Server Key Exchange message. This meets the test requirement.

7.4.9 FCS_TLSC_EXT.1.1 TEST #6A

Item	Data
Test Assurance Activity	Modify a byte in the Server Finished handshake message and verify that the handshake does not finish successfully and no application data flows.
Test Steps	• Start the server using the 'acumen-tls' tool and verify the connection when a byte is modified in the server finished handshake. • Verify the error logs on the device showing the digest check failed. • Verify the unsuccessful connection with packet capture.
Expected Test Results	• TOE should reject a connection when the tool modifies the server finished handshake message. • TOE logs should show connection failure due to the digest check failed. • Packet capture should show encrypted alert is generated by TOE and the handshake does not finish successfully.
Pass/Fail with Explanation	Pass. When a byte is modified in the Server Finished handshake message the handshake does not finish successfully and no application data flows. This meets the test requirements.

7.4.10 FCS_TLSC_EXT.1.1 TEST #6B

Item	Data
Test Assurance Activity	Send a garbled message from the server after the server has issued the ChangeCipherSpec message and verify that the handshake does not finish successfully and no application data flows.
Test Steps	• Start the server using the 'acumen-tlsc-v2.2e' tool and verify the connection when a garbled message is sent after the Change CipherSpec message. • Verify the error logs on the device showing data received between ChangeCipherSpec (CCS) message and finished. • Verify the unsuccessful connection with packet capture.
Expected Test Results	• Handshake should not happen when TOE receives a garbled message. • TOE logs should show connection failure due to data received between ChangeCipherSpec (CCS) message and finished. • Packet capture should an encrypted alert is generated by TOE as the garbled message is sent after the ChangeCipherSpec message.

Pass/Fail with Explanation	Pass. The TOE rejects the connection after receiving garbled data after the ChangeCipherSpec message. This meets the test requirements.

7.4.11 FCS_TLSC_EXT.1.1 TEST #6C

Item	Data
Test Assurance Activity	Modify at least one byte in the server's nonce in the Server Hello handshake message and verify that the client rejects the Server Key Exchange handshake message (if using a DHE or ECDHE ciphersuite) or that the server denies the client's Finished handshake message.
Test Steps	• Start the server using the 'acumen-tls' tool and verify the connection when a byte is modified in the server's nonce in the Server Hello handshake message. • Verify the error logs showing handshake failure due to a bad signature. • Verify the unsuccessful connection with packet capture.
Expected Test Results	• Client should reject the handshake message when nonce in the server hello handshake is changed. • TOE logs should show handshake failure due to a bad signature. • Packet capture should show a fatal error generated by TOE because bytes are modified in server nonce.
Pass/Fail with Explanation	Pass. TOE rejects the connection when the byte is modified in the server's nonce in the Server Hello handshake message. This meets the test requirements.

7.4.12 FCS_TLSC_EXT.1.2 TEST #1

Item	Data
Test Assurance Activity	This test is applicable if **TLS-based communications with RFC 6125 is selected for FTP_ITC.1, FTP_TRP, or FPT_ITT.** The evaluator shall present a server certificate that contains a CN that does not match the reference identifier and does not contain the SAN extension. The evaluator shall verify that the connection fails.

	The evaluator shall repeat this test for each identifier type (e.g. IPv4, IPv6, FQDN) supported in the CN. When testing IPv4 or IPv6 addresses, the evaluator shall modify a single decimal or hexadecimal digit in the CN. Remark: Some systems might require the presence of the SAN extension. In this case the connection would still fail but for the reason of the missing SAN extension instead of the mismatch of CN and reference identifier. Both reasons are acceptable to pass Test 1.
Test Steps	CN as IPV4: • Configure the TOE for reference identifier name as IPV4. • Configure the Server certificate showing invalid CN. • Configure the Server certificate showing no SAN extension. • Establish a connection with the TOE over TLS and verify the connection failure. • Verify the connection failure logs on the device that state 'certificate verify failed'. • Verify the unsuccessful connection due to an invalid CN in the packet capture. CN as IPV6: • Configure the TOE for reference identifier name as IPV6. • Configure the Server certificate showing invalid CN. • Configure the Server certificate showing no SAN extension. • Establish a connection with the TOE over TLS and verify the connection failure. • Verify the connection failure logs on the device that state 'certificate verify failed'. • Verify the unsuccessful connection due to an invalid CN in the packet capture. CN as FQDN: • Configure the TOE for reference identifier name as FQDN. • Configure the Server certificate showing invalid CN. • Configure the Server certificate showing no SAN extension. • Establish a connection with the TOE over TLS and verify the connection failure. • Verify the connection failure logs on the device that state 'certificate verify failed'. • Verify the unsuccessful connection due to an invalid CN in a packet capture.

Expected Test Results	• The TOE should reject certificates with an invalid CN and No SAN. • TOE logs should show connection failure due to invalid CN and No SAN. • Packet capture should show invalid CN and no SAN is configured in the certificate and FIN message is generated by TOE.
Pass/Fail with Explanation	Pass. The TOE rejects connection when a server certificate that contains a CN that does not match the reference identifier type for IPv4, IPv6, or FQDN and does not contain the SAN extension is presented. This meets the testing requirements.

7.4.13 FCS_TLSC_EXT.1.2 TEST #2

Item	Data
Test Assurance Activity	This test is applicable if **TLS-based communications with RFC 6125 is selected for FTP_ITC.1, FTP_TRP, or FPT_ITT.** The evaluator shall present a server certificate that contains a CN that matches the reference identifier, contains the SAN extension, but does not contain an identifier in the SAN that matches the reference identifier. The evaluator shall verify that the connection fails. The evaluator shall repeat this test for each supported SAN type (e.g. IPv4, IPv6, FQDN, URI). When testing IPv4 or IPv6 addresses, the evaluator shall modify a single decimal or hexadecimal digit in the SAN.
Test Steps	CN and SAN as IPV4: • Configure the TOE for reference identifier name as IPV4. • Configure the Server certificate showing valid CN. • Configure the Server certificate showing an invalid SAN. • Initiate the connection from the TOE to the TLS Server and verify the connection failure. • Verify the connection failure logs on the device that state 'certificate verify failed'. • Verify the unsuccessful connection due to a valid CN but an invalid SAN via a packet capture. CN and SAN as IPV6: • Configure the TOE for reference identifier name as IPV6. • Configure the Server certificate showing valid CN. • Configure the Server certificate showing an invalid SAN.

	• Initiate the connection from the TOE to the TLS Server and verify the connection failure. • Verify the connection failure logs on the device that state 'certificate verify failed'. • Verify the unsuccessful connection due to a valid CN but an invalid SAN via a packet capture. CN and SAN as FQDN: • Configure the TOE for reference identifier name as FQDN. • Configure the Server certificate showing valid CN. • Configure the Server certificate showing an invalid SAN. • Initiate the connection from the TOE to the TLS Server and verify the connection failure. • Verify the connection failure logs on the device that state 'certificate verify failed'. • Verify the unsuccessful connection due to a valid CN but an invalid SAN via a packet capture.
Expected Test Results	• The TOE should reject certificates with a correct CN but incorrect SAN. • TOE logs should show connection failure due to SAN mismatch. • Packet capture should show valid CN and invalid SAN in configured in the certificate and FIN message is generated by TOE.
Pass/Fail with Explanation	Pass. The TOE rejects the connection when a server certificate contains a CN that matches the reference identifier type for IPv4, IPv6, or FQDN in the CN field but contains an invalid SAN extension. This meets the testing requirements.

7.4.14 FCS_TLSC_EXT.1.2 TEST #3

Item	Data
Test Assurance Activity	This test is applicable if **TLS-based communications with RFC 6125 is selected for FTP_ITC.1, FTP_TRP, or FPT_ITT.** If the TOE does not mandate the presence of the SAN extension, the evaluator shall present a server certificate that contains a CN that matches the reference identifier and does not contain the SAN extension. The evaluator shall verify that the connection succeeds. The evaluator shall repeat this test for each identifier type (e.g. IPv4, IPv6, FQDN) supported in the CN. If the TOE does mandate the presence of the SAN extension, this Test shall be omitted.

Test Steps	The TOE mandates the presence of the SAN extension when the reference identifier is an IPv4 or IPv6 address but does not mandate it when the reference identifier is an FQDN. Therefore, for this test, the FQDN will be tested. CN: FQDN • Configure the correct reference identifier in the TOE. • Create a server certificate with valid CN but no SAN. • Connect to the TLS Server and verify that the connection is established. • Verify successful connection with packet capture.
Expected Test Results	• The TOE should accept the connection when the certificate with a valid CN and No SAN is presented. • Packet capture should show a successful connection with a valid CN.
Pass/Fail with Explanation	Pass. The TOE successfully accepts the connection when a server certificate is presented with a CN matching the reference identifier as an FQDN in the CN field, even if the SAN extension is not included, as it does not mandate the SAN extension for FQDN reference identifiers. However, the TOE mandates the presence of the SAN extension when the reference identifier is an IPv4 or IPv6 address, so these tests are omitted. This meets the testing requirements.

7.4.15 FCS_TLSC_EXT.1.2 TEST #4

Item	Data
Test Assurance Activity	This test is applicable if **TLS-based communications with RFC 6125 is selected for FTP_ITC.1, FTP_TRP, or FPT_ITT.** The evaluator shall present a server certificate that contains a CN that does not match the reference identifier but does contain an identifier in the SAN that matches. The evaluator shall verify that the connection succeeds. The evaluator shall repeat this test for each supported SAN type (e.g. IPv4, IPv6, FQDN, SRV).
Test Steps	CN and SAN as IPV4: • Configure the TOE for reference identifier name as IPV4. • Configure the Server certificate showing invalid CN. • Configure the Server certificate showing a valid SAN extension. • Establish a connection with the TOE over TLS and verify the successful connection.

	• Verify through packet capture, that when a server certificate contains a CN that does not match the reference identifier but does contain an identifier in the SAN that matches, the connection is successfully established. CN and SAN as IPV6: • Configure the TOE for reference identifier name as IPV6. • Configure the Server certificate showing invalid CN. • Configure the Server certificate showing a valid SAN extension. • Establish a connection with the TOE over TLS and verify the successful connection. • Verify through packet capture, that when a server certificate contains a CN that does not match the reference identifier but does contain an identifier in the SAN that matches, the connection is successfully established. CN and SAN as FQDN: • Configure the TOE for reference identifier name as FQDN. • Configure the Server certificate showing invalid CN. • Configure the Server certificate showing a valid SAN extension. • Establish a connection with the TOE over TLS and verify the successful connection. • Verify through packet capture, that when a server certificate contains a CN that does not match the reference identifier but does contain an identifier in the SAN that matches, the connection is successfully established.
Expected Test Results	• The TOE should accept the connection when the certificate with an invalid CN and valid SAN is presented. • Packet capture should show a successful connection when the certificate with an invalid CN and valid SAN is presented.
Pass/Fail with Explanation	Pass. The TOE successfully accepts the connection when the server certificate that contains a CN that does not match the reference identifier but does contain an identifier type for IPv4, IPv6, or FQDN in the CN field in the SAN that matches is presented. This meets the testing requirements.

7.4.16 FCS_TLSC_EXT.1.2 TEST #5 (1)

Item	Data

Test Assurance Activity	This test is applicable if **TLS-based communications with RFC 6125 is selected for FTP_ITC.1, FTP_TRP, or FPT_ITT.** Test 5: The evaluator shall perform the following wildcard tests with each supported type of reference identifier that includes a DNS name (i.e. CN-ID with DNS, DNS-ID, SRV-ID, URI-ID): The evaluator shall present a server certificate containing a wildcard that is not in the left-most label of the presented identifier (e.g. foo.*.example.com) and verify that the connection fails.
Test Steps	CN: • Configure the TOE for the correct reference identifier. • Configure the server certificate showing a wildcard that is not in the left-most label of CN. • Establish a connection with the TOE over TLS and verify the unsuccessful connection. • Verify the error logs on the device. • Verify the unsuccessful connection with packet capture. SAN: • Configure the TOE for the correct reference identifier. • Configure the server certificate showing a wildcard that is not in the left-most label of SAN. • Establish a connection with the TOE over TLS and verify the unsuccessful connection. • Verify the error logs on the device. • Verify the unsuccessful connection with packet capture.
Expected Test Results	• TOE should reject the connection when the reference identifier does not match the presented wildcard which is not in the leftmost label. • TOE logs should show connection failure due to CN/SAN mismatch. • Packet capture should show that the FIN message is generated by TOE due to mismatched parameters.
Pass/Fail with Explanation	Pass. TOE rejects the connection when the reference identifier does not match the presented wildcard which is not in the leftmost label. This meets the testing requirements.

7.4.17 FCS_TLSC_EXT.1.2 TEST #5 (2)(A)

Item	Data

Test Assurance Activity	This test is applicable if **TLS-based communications with RFC 6125 is selected for FTP_ITC.1, FTP_TRP, or FPT_ITT.** Test 5: The evaluator shall perform the following wildcard tests with each supported type of reference identifier that includes a DNS name (i.e. CN-ID with DNS, DNS-ID, SRV-ID, URI-ID): The evaluator shall present a server certificate containing a wildcard in the left-most label (e.g. *.example.com). The evaluator shall configure the reference identifier with a single left-most label (e.g. foo.example.com) and verify that the connection succeeds, if wildcards are supported, or fails if wildcards are not supported. (Remark: Support for wildcards was always intended to be optional. It is sufficient to state that the TOE does not support wildcards and observe rejected connection attempts to satisfy corresponding assurance activities.)
Test Steps	CN: • Configure the TOE for the reference identifier with a single left-most label. • Configure the server certificate showing a wildcard in the leftmost label in CN. • Establish a connection with the TOE over TLS and verify the successful connection. • Verify the successful connection via packet capture. SAN: • Configure the TOE for the reference identifier with a single left-most label. • Configure the server certificate showing a wildcard in the leftmost label in SAN. • Establish a connection with the TOE over TLS and verify the successful connection. • Verify the successful connection via packet capture.
Expected Test Results	• TOE should accept the connection when the reference identifier with single left-most labels is presented in the certificate. • Packet capture should show a successful connection.
Pass/Fail with Explanation	Pass. TOE accepts the connection when the reference identifier with single left-most labels is presented in the certificate. This meets the testing requirements.

7.4.18 FCS_TLSC_EXT.1.2 TEST #5 (2)(B)

Item	Data
Test Assurance Activity	This test is applicable if **TLS-based communications with RFC 6125 is selected for FTP_ITC.1, FTP_TRP, or FPT_ITT.** Test 5: The evaluator shall perform the following wildcard tests with each supported type of reference identifier that includes a DNS name (i.e. CN-ID with DNS, DNS-ID, SRV-ID, URI-ID): The evaluator shall present a server certificate containing a wildcard in the left-most label (e.g. *.example.com). The evaluator shall configure the reference identifier without a left-most label as in the certificate (e.g. example.com) and verify that the connection fails. (Remark: Support for wildcards was always intended to be optional. It is sufficient to state that the TOE does not support wildcards and observe rejected connection attempts to satisfy corresponding assurance activities.)
Test Steps	CN: • Configure the TOE for the reference identifier without a leftmost label. • Configure the server certificate showing a wildcard in the leftmost label in CN. • Establish a connection with the TOE over TLS and verify the unsuccessful connection. • Verify the error logs on the device. • Verify the unsuccessful connection with packet capture. SAN: • Configure the TOE for the reference identifier without a leftmost label. • Configure the server certificate showing a wildcard in the leftmost label in SAN. • Establish a connection with the TOE over TLS and verify the unsuccessful connection. • Verify the error logs on the device. • Verify the unsuccessful connection with packet capture.
Expected Test Results	• When a server certificate containing a wildcard in the left-most label is presented, and the reference identifier without the left-most label configured as in the certificate, the connection should fail. • TOE logs should show connection failure due to CN/SAN mismatch. • Packet capture should show FIN message is generated by TOE due to mismatched parameters.

Pass/Fail with Explanation	Pass. When a server certificate containing a wildcard in the left-most label is presented, and the reference identifier without the left-most label configured as in the certificate, the connection fails. This meets the testing requirements.

7.4.19 FCS_TLSC_EXT.1.2 TEST #5 (2)(C)

Item	Data
Test Assurance Activity	This test is applicable if **TLS-based communications with RFC 6125 is selected for FTP_ITC.1, FTP_TRP, or FPT_ITT.** Test 5: The evaluator shall perform the following wildcard tests with each supported type of reference identifier that includes a DNS name (i.e. CN-ID with DNS, DNS-ID, SRV-ID, URI-ID): The evaluator shall present a server certificate containing a wildcard in the left-most label (e.g. *.example.com). The evaluator shall configure the reference identifier with two left-most labels (e.g. bar.foo.example.com) and verify that the connection fails. (Remark: Support for wildcards was always intended to be optional. It is sufficient to state that the TOE does not support wildcards and observe rejected connection attempts to satisfy corresponding assurance activities.)
Test Steps	CN: • Configure the TOE for the reference identifier with two leftmost labels. • Configure the server certificate showing a wildcard in the leftmost label in CN. • Establish a connection with the TOE over TLS and verify the unsuccessful connection. • Verify the failure logs on the TOE. • Verify the unsuccessful connection via packet capture. SAN: • Configure the TOE for the reference identifier with two leftmost labels. • Configure the server certificate showing a wildcard in the leftmost label in SAN. • Establish a connection with the TOE over TLS and verify the unsuccessful connection. • Verify the failure logs on the TOE.

	• Verify the unsuccessful connection via packet capture.
Expected Test Results	• When configured with a reference identifier with two left-most labels, the TOE should reject the connection when presented with a server certificate containing a wildcard in the left-most label. • TOE logs should show connection failure due to CN/SAN mismatch. • Packet capture should show that the FIN message is generated by TOE due to mismatched parameters.
Pass/Fail with Explanation	Pass. When configured with a reference identifier with two left-most labels, the TOE rejects the connection when presented with a server certificate containing a wildcard in the left-most label. This meets the testing requirements.

7.4.20 FCS_TLSC_EXT.1.2 TEST #6 [**TD0790]**

Item	Data
Test Assurance Activity	This test is applicable if **TLS-based communications with RFC 6125 is selected for FTP_ITC.1, FTP_TRP, or FPT_ITT**. The objective of this test is to ensure the TOE is able to differentiate between IP address identifiers that are not allowed to contain wildcards and other types of identifiers that may contain wildcards. Test 6:[conditional] If IP address identifiers are supported in the SAN or CN, the evaluator shall present a server certificate that contains a CN that matches the reference identifier, except one of the groups has been replaced with a wildcard asterisk (*) (e.g. CN=*.168.0.1 when connecting to 192.168.0.1, CN=2001:0DB8:0000:0000:0008:0800:200C:* when connecting to 2001:0DB8:0000:0000:0008:0800:200C:417A). The certificate shall not contain the SAN extension. The evaluator shall verify that the connection fails. The evaluator shall repeat this test for each supported IP address version (e.g. IPv4, IPv6). Remark: Some systems might require the presence of the SAN extension. In this case the connection would still fail but for the reason of the missing SAN extension instead of the mismatch of CN and reference identifier. Both reasons are acceptable to pass Test 6.
Test Steps	IPv4: • Configure the TOE for the correct reference identifier.

	• Create a server certificate without the SAN and with a CN that matches the reference identifier but replace one of the groups with an *. • Establish a connection with the TOE over TLS and verify the unsuccessful connection . • Verify the certificate validation failure logs on the device. • Verify the unsuccessful connection with packet capture. IPv6: • Configure the TOE for the correct reference identifier. • Create a server certificate without the SAN and with a CN that matches the reference identifier but replace one of the groups with an *. • Establish a connection with the TOE over TLS and verify the unsuccessful connection. • Verify the certificate validation failure logs on the device. • Verify the unsuccessful connection with packet capture.
Expected Test Results	• TOE should reject the connection when the configured server certificate has a missing SAN extension and contains a CN that matches the reference identifier IP with one of the groups replaced with an asterisk (*). • TOE logs should generate certificate validation failure logs. • Packet capture should show failure due to CN mismatch.
Pass/Fail with Explanation	Pass. TOE rejects the connection when the configured server certificate has a missing SAN extension and contains a CN that matches the reference identifier IP(IPv4 and IPv6) with one of the groups replaced with an asterisk (*). This meets the test requirements

7.4.21 FCS_TLSC_EXT.1.2 TEST #7A

Item	Data
Test Assurance Activity	If **the secure channel is used for FPT_ITT, and RFC 5280 is selected**, the evaluator shall perform the following tests. Note, when multiple attribute types are selected in the SFR (e.g. when multiple attribute types are combined to form the unique identifier), the evaluator modifies each attribute type in accordance with the matching criteria described in the TSS (e.g. creating a mismatch of one attribute type at a time while other attribute types contain values that will match a portion of the reference identifier):

	The evaluator shall present a server certificate that does not contain an identifier in the Subject (DN) attribute type(s) that matches the reference identifier. The evaluator shall verify that the connection fails.
Pass/Fail with Explanation	N/A. This is not applicable as the secure channel is not used for FPT_ITT, and RFC 5280 is not selected in ST.

7.4.22 FCS_TLSC_EXT.1.2 TEST #7B

Item	Data
Test Assurance Activity	If **the secure channel is used for FPT_ITT, and RFC 5280 is selected**, the evaluator shall perform the following tests. Note, when multiple attribute types are selected in the SFR (e.g. when multiple attribute types are combined to form the unique identifier), the evaluator modifies each attribute type in accordance with the matching criteria described in the TSS (e.g. creating a mismatch of one attribute type at a time while other attribute types contain values that will match a portion of the reference identifier): The evaluator shall present a server certificate that contains a valid identifier as an attribute type other than the expected attribute type (e.g. if the TOE is configured to expect id-at-serialNumber=correct_identifier, the certificate could instead include id-at-name=correct_identifier), and does not contain the SAN extension. The evaluator shall verify that the connection fails. Remark: Some systems might require the presence of the SAN extension. In this case the connection would still fail but for the reason of the missing SAN extension instead of the mismatch of CN and reference identifier. Both reasons are acceptable to pass this test.
Pass/Fail with Explanation	N/A. This is not applicable as the secure channel is not used for FPT_ITT, and RFC 5280 is not selected in ST.

7.4.23 FCS_TLSC_EXT.1.2 TEST #7C

Item	Data

Test Assurance Activity	If **the secure channel is used for FPT_ITT, and RFC 5280 is selected**, the evaluator shall perform the following tests. Note, when multiple attribute types are selected in the SFR (e.g. when multiple attribute types are combined to form the unique identifier), the evaluator modifies each attribute type in accordance with the matching criteria described in the TSS (e.g. creating a mismatch of one attribute type at a time while other attribute types contain values that will match a portion of the reference identifier): The evaluator shall present a server certificate that contains a Subject attribute type that matches the reference identifier and does not contain the SAN extension. The evaluator shall verify that the connection succeeds.
Pass/Fail with Explanation	N/A. This is not applicable as the secure channel is not used for FPT_ITT, and RFC 5280 is not selected in ST.

7.4.24 FCS_TLSC_EXT.1.2 TEST #7D

Item	Data
Test Assurance Activity	If **the secure channel is used for FPT_ITT, and RFC 5280 is selected**, the evaluator shall perform the following tests. Note, when multiple attribute types are selected in the SFR (e.g. when multiple attribute types are combined to form the unique identifier), the evaluator modifies each attribute type in accordance with the matching criteria described in the TSS (e.g. creating a mismatch of one attribute type at a time while other attribute types contain values that will match a portion of the reference identifier): The evaluator shall confirm that all use of wildcards results in connection failure regardless of whether the wildcards are used in the left or right side of the presented identifier. (Remark: Use of wildcards is not addressed within RFC 5280.)
Pass/Fail with Explanation	N/A. This is not applicable as the secure channel is not used for FPT_ITT, and RFC 5280 is not selected in ST.

7.4.25 FCS_TLSC_EXT.1.3 TEST #1

Item	Data
Test Assurance Activity	Using the administrative guidance, the evaluator shall load a CA certificate or certificates needed to validate the presented certificate used to authenticate an external entity and demonstrate that the function succeeds, and a trusted channel can be established.
Test Steps	• Configure TOE to connect to the TLS server. • Create a complete chain of certificates. • Upload a complete certificate validation chain to the TOE. • Attempt the connection from the TOE to the TLS server and verify the successful connection (complete certificate chain present). • Verify the successful connection with packet capture.
Expected Test Results	• When a complete certificate trust chain is present, the TOE should successfully establish a connection with the TLS server. • Packet capture should show a successful connection.
Pass/Fail with Explanation	Pass. When a complete certificate trust chain is present, the TOE successfully establishes a connection with the TLS server. This meets the test requirements.

7.4.26 FCS_TLSC_EXT.1.3 TEST #2

Item	Data
Test Assurance Activity	The evaluator shall then change the presented certificate(s) so that validation fails and show that the certificate is not automatically accepted. The evaluator shall repeat this test to cover the selected types of failure defined in the SFR (i.e. the selected ones from failed matching of the reference identifier, failed validation of the certificate path, failed validation of the expiration date, failed determination of the revocation status). The evaluator performs the action indicated in the SFR selection observing the TSF resulting in the expected state for the trusted channel (e.g. trusted channel was established) covering the types of failure for which an override mechanism is defined.
Test Steps	Failed matching reference Identifier:

	• The requirements of this test case are exercised in FCS_TLSC_EXT.1.2 Test #1 and Test #2. Failed validation of the certificate path: • Remove the ICA from a chain on the TOE. • Establish a connection with the TOE over TLS and verify that it fails. • Verify the failure logs on the device, showing the 'certificate verify failed'. • Verify the unsuccessful connection with packet capture. Failed validation of the expiration date: • Create a server certificate that is expired. • Show the clock on the TOE. • Establish a connection with the TOE over TLS and verify that it fails. • Verify the failure logs on the device, showing connection is not established due to an expired certificate. • Verify the unsuccessful connection with packet capture. Failed determination of the revocation status • The requirements of this test case are exercised in FIA_X509_EXT.2 Test #1.
Expected Test Results	• The TOE should reject the Invalid certificates. • TOE logs and packet capture should show an error while connecting to the TLS server.
Pass/Fail with Explanation	Pass. Failed matching of the reference identifier test is covered by FCS_TLSC_EXT.1.2 Test #1 and Test #2, The TOE rejects the connection when an incomplete certificate trust chain is present, The TOE rejects the connection when an expired certificate is used, and Failed determination of revocation status test is covered by FIA_X509_EXT.2 Test #1.

7.4.27 FCS_TLSC_EXT.1.3 TEST #3

Item	Data
Test Assurance Activity	The purpose of this test to verify that only selected certificate validation failures could be administratively overridden. **If any override mechanism is defined for failed certificate validation**, the evaluator shall configure a new presented certificate that does not contain a valid entry in one of the mandatory fields or

	parameters (e.g. inappropriate value in extendedKeyUsage field) but is otherwise valid and signed by a trusted CA. The evaluator shall confirm that the certificate validation fails (i.e. certificate is rejected), and there is no administrative override available to accept such certificate.
Pass/Fail with Explanation	N/A. This test is not applicable as TOE does not implement any administrator override mechanism as per ST.

7.4.28 FCS_TLSC_EXT.1.4 TEST #1

Item	**Data**
Test Assurance Activity	If the TOE presents the **Supported Elliptic Curves/Supported Groups Extension**, the evaluator shall configure the server to perform ECDHE or DHE (as applicable) key exchange using each of the TOE's supported curves and/or groups. The evaluator shall verify that the TOE successfully connects to the server.
Test Steps	• Initiate the connection from the TOE to the TLS Server using the curve secp256r1 and verify the successful connection. • Verify with packet capture that the required curve is secp256r1. • Initiate the connection from the TOE to the TLS Server using the curve secp384r1 and verify the successful connection. • Verify with packet capture that the required curve is secp384r1. • Initiate the connection from the TOE to the TLS Server using the curve secp521r1 and verify the successful connection. • Verify with packet capture that the required curve is secp521r1.
Expected Test Results	• The TOE should establish a connection successfully when the supported curves are presented. • Packet capture shows a successful connection and elliptic curved used.
Pass/Fail with Explanation	Pass. The TOE successfully established a connection with the TLS server when supported curves were introduced. This meets the test requirements.

7.5 TLSS[30]

7.5.1 FCS_TLSS_EXT.1.1 TEST #1

Item	Data
Test Assurance Activity	Test 1: The evaluator shall establish a TLS connection using each of the ciphersuites specified by the requirement. This connection may be established as part of the establishment of a higher-level protocol, e.g., as part of an HTTPS session. It is sufficient to observe the successful negotiation of a ciphersuite to satisfy the intent of the test; it is not necessary to examine the characteristics of the encrypted traffic to discern the ciphersuite being used (for example, that the cryptographic algorithm is 128-bit AES and not 256-bit AES).
Test Steps	• Establish a connection with the TOE over TLS using the ciphersuite TLS_DHE_RSA_WITH_AES_128_CBC_SHA. • Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_DHE_RSA_WITH_AES_256_CBC_SHA. • Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256. • Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256. • Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256. • Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384. • Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256. • Verify the required ciphersuite with packet capture.

[30]VX series models doesn't support Web UI Feature and hence this selection-based SFR is not applicable to the VX Series Models

	• Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384. • Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256. • Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384. • Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256. • Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384. • Verify the required ciphersuite with packet capture.
Expected Test Results	• TOE should successfully establish the TLS connection with claimed ciphersuites. • Packet captures should show the successful establishment of TLS connection with configured ciphersuites.
Pass/Fail with Explanation	Pass. The TOE was able to make the successful connection via the supported ciphersuites. This meets the testing requirements.

7.5.2 FCS_TLSS_EXT.1.1 TEST #2

Item	Data
Test Assurance Activity	Test 2: The evaluator shall send a Client Hello to the server with a list of ciphersuites that does not contain any of the ciphersuites in the server's ST and verify that the server denies the connection. Additionally, the evaluator shall send a Client Hello to the server containing only the TLS_NULL_WITH_NULL_NULL ciphersuite and verify that the server denies the connection.
Test Steps	• Using the 'acumen-tlss-v2.2e' tool as a client, attempt to establish a TLS connection to the TOE using an unsupported ciphersuite in the Client Hello:

	TLS_RSA_WITH_NULL_MD5 • Verify the logs on TOE showing handshake failure. • Verify the connection fails via packet capture. • Using the 'acumen-tlss-v2.2e' tool as a client, attempt to establish a TLS connection to the TOE using. TLS_NULL_WITH_NULL_NULL ciphersuite in the client hello and verify the connection fails. • Verify the logs on TOE showing handshake failure. • Verify the connection fails via packet capture.
Expected Test Results	• Connection should be rejected when the unsupported ciphersuite is present. • Packet capture should show handshake failure with unsupported ciphersuites. • The log on TOE should show handshake failure.
Pass/Fail with Explanation	Pass. The TOE rejects TLS connections with the unsupported ciphersuites. This meets the testing requirement.

7.5.3 FCS_TLSS_EXT.1.1 TEST #3A

Item	Data
Test Assurance Activity	Modify a byte in the Client Finished handshake message, and verify that the server rejects the connection and does not send any application data.
Test Steps	• Run the 'acumen-tls' tool as a client with a modified client finished message and wait for the connection, the connection should fail. • Verify the logs on TOE showing handshake failure. • Verify the unsuccessful connection via packet capture.
Expected Test Results	• TOE should reject a connection when the byte in the client's finished handshake message is modified. • Packet capture should show connection failure when the Client Finished handshake message is modified. • TOE logs should show handshake failure.
Pass/Fail with Explanation	Pass. The TOE rejects the connection after receiving the modified Client Handshake message. This meets the testing requirements.

7.5.4 FCS_TLSS_EXT.1.1 TEST #3B

Item	Data
Test Assurance Activity	(Test Intent: The intent of this test is to ensure that the server's TLS implementation immediately makes use of the key exchange and authentication algorithms to: a) Correctly encrypt (D)TLS Finished message and b) Encrypt every (D)TLS message after session keys are negotiated.) The evaluator shall use one of the claimed ciphersuites to complete a successful handshake and observe transmission of properly encrypted application data. The evaluator shall verify that no Alert with alert level Fatal (2) messages were sent. The evaluator shall verify that the Finished message (Content type hexadecimal 16 and handshake message type hexadecimal 14) is sent immediately after the server's ChangeCipherSpec (Content type hexadecimal 14) message. The evaluator shall examine the Finished message (encrypted example in hexadecimal of a TLS record containing a Finished message, 16 03 03 00 40 11 22 33 44 55...) and confirm that it does not contain unencrypted data (unencrypted example in hexadecimal of a TLS record containing a Finished message, 16 03 03 00 40 14 00 00 0c...), by verifying that the first byte of the encrypted Finished message does not equal hexadecimal 14 for at least one of three test messages. There is a chance that an encrypted Finished message contains a hexadecimal value of '14' at the position where a plaintext Finished message would contain the message type code '14'. If the observed Finished message contains a hexadecimal value of '14' at the position where the plaintext Finished message would contain the message type code, the test shall be repeated three times in total. In case the value of '14' can be observed in all three tests it can be assumed that the Finished message has indeed been sent in plaintext and the test has to be regarded as 'failed'. Otherwise it has to be assumed that the observation of the value '14' has been due to chance and that the Finished message has indeed been sent encrypted. In that latter case the test shall be regarded as 'passed'.
Test Steps	• Initiate a connection to the TOE with the ‘acumen-tlss-v2.2e’ tool as a client. • Verify that no Alert with alert level Fatal (2) messages were sent.

	• Verify that the Finished message (Content type hexadecimal 16 and handshake message type hexadecimal 14) is sent immediately after the server's ChangeCipherSpec (Content type hexadecimal 14) message. • Examine the Finished message and confirm that it does not contain unencrypted data by verifying that the first byte of the encrypted Finished message does not equal hexadecimal 14 for at least one of three test messages.
Expected Test Results	• TOE should reject a connection when text is not encrypted otherwise it should succeed. • Evidence (Packet capture) showing the message is encrypted hence the connection is successful.
Pass/Fail with Explanation	Pass. No Alert with alert level Fatal (2) messages were sent. The Finished message contains Hexadecimal 16 and is sent immediately after Hexadecimal 14 in the ChangeCipherSpec message. The first byte of the encrypted Finished message does not equal hexadecimal 14. This meets the testing requirement.

7.5.5 FCS_TLSS_EXT.1.2 TEST #1

Item	Data
Test Assurance Activity	The evaluator shall send a Client Hello requesting a connection for all mandatory and selected protocol versions in the SFR (e.g. by enumeration of protocol versions in a test client) and verify that the server denies the connection for each attempt.
Test Steps	• Use the 'acumen-tlss-v2.2e' tool as a client to initiate a connection to the TOE and verify the connection fails for all the non-supported SSL and TLS versions. • Verify the connection fails with SSLv2.0. • Verify the logs on TOE showing handshake failure. • Verify handshake failure using packet capture. • Verify the connection fails with SSLv3.0. • Verify the logs on TOE showing handshake failure. • Verify handshake failure using packet capture. • Verify the connection fails with TLSv1.0. • Verify the logs on TOE showing handshake failure. • Verify handshake failure using packet capture. • Verify the connection fails with TLSv1.1. • Verify the logs on TOE showing handshake failure. • Verify handshake failure using packet capture.

Expected Test Results	• The server should reject a connection when a client requests a connection with the unsupported TLS/SSL versions. • TOE logs should show connection failure due to an unknown protocol. • Packet capture should show a connection reset due to an unsupported protocol version.
Pass/Fail with Explanation	Pass. The TOE rejects all SSLv2, SSLv3, TLS v1.0, and TLS v1.1 connection attempts. This meets the testing requirement.

7.5.6 FCS_TLSS_EXT.1.3 TEST #1A

Item	Data
Test Assurance Activity	If **ECDHE ciphersuites** are supported: The evaluator shall repeat this test for each supported elliptic curve. The evaluator shall attempt a connection using a supported ECDHE ciphersuite and a single supported elliptic curve specified in the Elliptic Curves Extension. The Evaluator shall verify (though a packet capture or instrumented client) that the TOE selects the same curve in the Server Key Exchange message and successfully establishes the connection.
Test Steps	• Initiate a connection with the TOE over TLS using the curve secp256r1 and verify the connection is successful. • Verify the packet capture showing the curve secp256r1. • Initiate a connection with the TOE over TLS using the curve secp384r1 and verify the connection is successful. • Verify the packet capture showing the curve secp384r1. • Initiate a connection with the TOE over TLS using the curve secp521r1 and verify the connection is successful. • Verify the packet capture showing the curve secp521r1.
Expected Test Results	• The connection should be successful when a supported ECDHE cipher and elliptic curve are configured. • Packet capture should show a successful connection and the supported elliptic curve used.
Pass/Fail with Explanation	Pass. The TOE was able to make the connection using each supported elliptic curve. This meets the testing requirements.

7.5.7 FCS_TLSS_EXT.1.3 TEST #1B

Item	Data
Test Assurance Activity	If **ECDHE ciphersuites** are supported: The evaluator shall attempt a connection using a supported ECDHE ciphersuite and a single unsupported elliptic curve (e.g. secp192r1 (0x13)) specified in RFC4492, chap. 5.1.1. The evaluator shall verify that the TOE does not send a Server Hello message and the connection is not successfully established.
Test Steps	• Run the 'acumen-tlss' tool as a client, establish a connection to TOE over TLS using the supported ciphersuite and unsupported elliptical curve, and verify the connection fails. • Verify the log on the device showing handshake failure. • Verify the packet capture showing connection failure.
Expected Test Results	• Connection should be rejected when supported cipher and the unsupported elliptic curve are configured. • Packet capture should show connection failure with the unsupported elliptic curve. • Logs showing handshake failure.
Pass/Fail with Explanation	Pass. The TOE rejects a connection with unsupported elliptic curves. This meets the testing requirements.

7.5.8 FCS_TLSS_EXT.1.3 TEST #2

Item	Data
Test Assurance Activity	If **DHE ciphersuites** are supported, the evaluator shall repeat the following test for each supported parameter size. If any configuration is necessary, the evaluator shall configure the TOE to use a supported Diffie-Hellman parameter size. The evaluator shall attempt a connection using a supported DHE ciphersuite. The evaluator shall verify (through a packet capture or instrumented client) that the TOE sends a Server Key Exchange Message where p Length is consistent with the message are the ones configured Diffie-Hellman parameter size(s).
Test Steps	• Connect to the TOE using DHE 2048 bits and verify that it is successful. • Verify with a packet capture showing the modulus that corresponds to the specified DHE ciphersuite.

Expected Test Results	• The TOE should establish a successful TLS connection with the supported DHE ciphersuite. • The packet capture should show the modulus that corresponds to the specified DHE ciphersuite.
Pass/Fail with Explanation	Pass. The TOE was able to establish the connection using the supported DH key. This meets the testing requirement.

7.5.9 FCS_TLSS_EXT.1.3 TEST #3

Item	Data
Test Assurance Activity	If **RSA key establishment ciphersuites** are supported, the evaluator shall repeat this test for each RSA key establishment key size. If any configuration is necessary, the evaluator shall configure the TOE to perform RSA key establishment using a supported key size (e.g. by loading a certificate with the appropriate key size). The evaluator shall attempt a connection using a supported RSA key establishment ciphersuite. The evaluator shall verify (through a packet capture or instrumented client) that the TOE sends a certificate whose modulus is consistent with the configured RSA key size.
Pass/Fail with Explanation	N/A. RSA key establishment is not selected in the ST.

7.5.10 FCS_TLSS_EXT.1.4 TEST #1 [**TD0569**]

Item	Data
Test Assurance Activity	If the **TOE does not support session resumption based on session IDs according to RFC4346 (TLS1.1) or RFC5246 (TLS1.2) or session tickets according to RFC5077**, the evaluator shall perform the following test: a) The client sends a Client Hello with a zero-length session identifier and with a SessionTicket extension containing a zero-length ticket. b) The client verifies the server does not send a NewSessionTicket handshake message (at any point in the handshake). c) The client verifies the Server Hello message contains a zero-length session identifier or passes the following steps:

	Note: The following steps are only performed if the ServerHello message contains a non-zero length SessionID. d) The client completes the TLS handshake and captures the SessionID from the ServerHello. e) The client sends a ClientHello containing the SessionID captured in step d). This can be done by keeping the TLS session in step d) open or start a new TLS session using the SessionID captured in step d). f) The client verifies the TOE: a. implicitly rejects the SessionID by sending a ServerHello containing a different SessionID and by performing a full handshake (as shown in Figure 1 of RFC 4346 or RFC 5246), or b. terminates the connection in some way that prevents the flow of application data. Remark: If multiple contexts are supported for session resumption, the session ID or session ticket may be obtained in one context for resumption in another context. It is possible that one or more contexts may only permit the construction of sessions to be reused in other contexts but not actually permit resumption themselves. For contexts which do not permit resumption, the evaluator is required to verify this behaviour subject to the description provided in the TSS. It is not mandated that the session establishment and session resumption share context. For example, it is acceptable for a control channel to establish and application channel to resume the session.
Pass/Fail with Explanation	N/A. The TOE supports session tickets.

7.5.11 FCS_TLSS_EXT.1.4 TEST #2A [**TD0569**]

Item	**Data**
Test Assurance Activity	If the **TOE supports session resumption using session IDs according to RFC4346 (TLS1.1) or RFC5246 (TLS1.2)**, the evaluator shall carry out the following steps (note that for each of these tests, it is not necessary to perform the test case for each supported version of TLS): The evaluator shall conduct a successful handshake and capture the TOE-generated session ID in the Server Hello message. The evaluator shall then

	initiate a new TLS connection and send the previously captured session ID to show that the TOE resumed the previous session by responding with ServerHello containing the same SessionID immediately followed by ChangeCipherSpec and Finished messages (as shown in Figure 2 of RFC 4346 or RFC 5246). Remark: If multiple contexts are supported for session resumption, for each of the above test cases, the session ID may be obtained in one context for resumption in another context. There is no requirement that the session ID be obtained and replayed within the same context subject to the description provided in the TSS. All contexts that can reuse a session ID constructed in another context must be tested. It is not mandated that the session establishment and session resumption share context. For example, it is acceptable for a control channel to establish and application channel to resume the session.
Pass/Fail with Explanation	N/A. The TOE does not support session resumption using session IDs according to RFC4346 (TLS1.1) or RFC5246 (TLS1.2).

7.5.12 FCS_TLSS_EXT.1.4 TEST #2B [**TD0569**]

Item	Data
Test Assurance Activity	If the **TOE supports session resumption using session IDs according to RFC4346 (TLS1.1) or RFC5246 (TLS1.2)**, the evaluator shall carry out the following steps (note that for each of these tests, it is not necessary to perform the test case for each supported version of TLS): The evaluator shall initiate a handshake and capture the TOE-generated session ID in the Server Hello message. The evaluator shall then, within the same handshake, generate or force an unencrypted fatal Alert message immediately before the client would otherwise send its ChangeCipherSpec message thereby disrupting the handshake. The evaluator shall then initiate a new Client Hello using the previously captured session ID, and verify that the server (1) implicitly rejects the session ID by sending a ServerHello containing a different SessionID and performing a full handshake (as shown in figure 1 of RFC 4346 or RFC 5246), or (2) terminates the connection in some way that prevents the flow of application data.

	Remark: If multiple contexts are supported for session resumption, for each of the above test cases, the session ID may be obtained in one context for resumption in another context. There is no requirement that the session ID be obtained and replayed within the same context subject to the description provided in the TSS. All contexts that can reuse a session ID constructed in another context must be tested. It is not mandated that the session establishment and session resumption share context. For example, it is acceptable for a control channel to establish and application channel to resume the session.
Pass/Fail with Explanation	N/A. The TOE does not support session resumption using session IDs according to RFC4346 (TLS1.1) or RFC5246 (TLS1.2).

7.5.13 FCS_TLSS_EXT.1.4 TEST #3A **[TD0556, TD0569]**

Item	Data
Test Assurance Activity	If the **TOE supports session tickets according to RFC5077**, the evaluator shall carry out the following steps (note that for each of these tests, it is not necessary to perform the test case for each supported version of TLS): The evaluator shall permit a successful TLS handshake to occur in which a session ticket is exchanged with the non-TOE client. The evaluator shall then attempt to correctly reuse the previous session by sending the session ticket in the ClientHello. The evaluator shall confirm that the TOE responds with an abbreviated handshake described in section 3.1 of RFC 5077 and illustrated with an example in figure 2. Of particular note: if the server successfully verifies the client's ticket, then it may renew the ticket by including a NewSessionTicket handshake message after the ServerHello in the abbreviated handshake (which is shown in figure 2). This is not required, however as further clarified in section 3.3 of RFC 5077. Remark: If multiple contexts are supported for session resumption, for each of the above test cases, the session ticket may be obtained in one context for resumption in another context. There is no requirement that the session ticket be obtained and replayed within the same context subject to the description provided in the TSS. All contexts that can reuse a session ticket constructed in another context must be tested. It is not mandated that the session establishment and session resumption share context. For example, it is

	acceptable for a control channel to establish and application channel to resume the session.
Test Steps	• Use the 'acumen-tlss' tool to connect to the TOE. • Verify packet capture contains two TLS handshakes with the TOE and the same session ticket is sent through the next session's client hello. • Verify logs are generated for successful and closed connections.
Expected Test Results	• The TOE should establish a successful TLS client connection when the session ticket of the previous session is sent in the ClientHello. • The packet capture should confirm that the same session ticket is sent in the subsequent session's client hello, thereby establishing a successful connection.
Pass/Fail with Explanation	Pass. The TOE responds with an abbreviated handshake when the session ticket is reused. This meets the testing requirements.

s

7.5.14 FCS_TLSS_EXT.1.4 TEST #3B [TD0569]

Item	Data
Test Assurance Activity	If the **TOE supports session tickets according to RFC5077**, the evaluator shall carry out the following steps (note that for each of these tests, it is not necessary to perform the test case for each supported version of TLS): The evaluator shall permit a successful TLS handshake to occur in which a session ticket is exchanged with the non-TOE client. The evaluator will then modify the session ticket and send it as part of a new Client Hello message. The evaluator shall confirm that the TOE either (1) implicitly rejects the session ticket by performing a full handshake (as shown in figure 3 or 4 of RFC 5077), or (2) terminates the connection in some way that prevents the flow of application data. Remark: If multiple contexts are supported for session resumption, for each of the above test cases, the session ticket may be obtained in one context for resumption in another context. There is no requirement that the session ticket be obtained and replayed within the same context subject to the description provided in the TSS. All contexts that can reuse a session ticket constructed in another context must be tested. It is not mandated that the session establishment and session resumption share context. For example, it is

	acceptable for a control channel to establish and application channel to resume the session.
Test Steps	• Use the 'acumen-tlss' tool to connect to the TOE. • Verify packet capture contains two TLS handshakes with the TOE. • Verify logs are generated for successful and closed connections.
Expected Test Results	• The TOE should close the TLS client connection that is established by the 'acumen-tlss' tool which sends the modified session ticket. • The packet capture will show TOE implicitly rejects the modified session ticket by performing a full handshake by sending a new session ticket and allowing the flow of application data.
Pass/Fail with Explanation	Pass. TOE implicitly rejects the modified session ticket by performing a full handshake by sending a new session ticket and allowing the flow of application data. This meets the testing requirements.

7.6 UPDATE

7.6.1 FPT_TST_EXT.1 TEST #1

Item	Data
Test Assurance Activity	It is expected that at least the following tests are performed: a) Verification of the integrity of the firmware and executable software of the TOE b) Verification of the correct operation of the cryptographic functions necessary to fulfil any of the SFRs. Although formal compliance is not mandated, the self-tests performed should aim for a level of confidence comparable to: a) [FIPS 140-2], chap. 4.9.1, Software/firmware integrity test for the verification of the integrity of the firmware and executable software. Note that the testing is not restricted to the cryptographic functions of the TOE. b) [FIPS 140-2], chap. 4.9.1, Cryptographic algorithm test for the verification of the correct operation of cryptographic functions. Alternatively, national requirements of any CCRA member state for the security evaluation of cryptographic functions should be considered as appropriate. The evaluator shall either verify that the self-tests described above are carried out during initial start-up or that the developer has justified any deviation from this. For distributed TOEs the evaluator shall perform testing of self-tests on all TOE components according to the description in the TSS about which self-test are performed by which component.
Test Steps	• Reboot the TOE. • Verify that the self tests were performed successfully.
Expected Test Results	• The TOE should execute all claimed self-tests during bootup. • Evidence (screenshot or CLI output) showing successful self-tests.

Pass/Fail with Explanation	Pass. The TOE successfully verifies the integrity of firmware and self-tests were performed correctly. This meets the testing requirement.

7.6.2 FPT_TUD_EXT.1 TEST #1

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 1: The evaluator performs the version verification activity to determine the current version of the product. If a trusted update can be installed on the TOE with a delayed activation, the evaluator shall also query the most recently installed version (for this test the TOE shall be in a state where these two versions match). The evaluator obtains a legitimate update using procedures described in the guidance documentation and verifies that it is successfully installed on the TOE. For some TOEs loading the update onto the TOE and activation of the update are separate steps ('activation' could be performed e.g. by a distinct activation step or by rebooting the device). In that case the evaluator verifies after loading the update onto the TOE but before activation of the update that the current version of the product did not change but the most recently installed version has changed to the new product version. After the update, the evaluator performs the version verification activity again to verify the version correctly corresponds to that of the update and that current version of the product and most recently installed version match again. The evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all methods supported (manual updates, automatic checking for updates, automatic updates). For distributed TOEs the evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all TOE components.
Test Steps	• Check the current image version on TOE. • Check the current active and inactive images on TOE. • Download the new image. • Verify the version of the downloaded image. • Install the new image. • Configure TOE to boot the new image. • Reload the TOE. • Check the new image version. • Verify successful image installation with logs.

Expected Test Results	• The TOE should successfully update the current version with the new version after verifying the integrity of the new image. • Evidence - screenshot showing new version post upgrade. • TOE logs should show successful image installation.
Pass/Fail with Explanation	Pass. The TOE can be successfully updated. This meets the testing requirements.

7.6.3 FPT_TUD_EXT.1 TEST #2 (A)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 2 [conditional]: If the TOE itself verifies a digital signature to authorize the installation of an image to update the TOE the following test shall be performed (otherwise the test shall be omitted). The evaluator first confirms that no updates are pending and then performs the version verification activity to determine the current version of the product, verifying that it is different from the version claimed in the update(s) to be used in this test. The evaluator obtains or produces illegitimate updates as defined below and attempts to install them on the TOE. The evaluator verifies that the TOE rejects all of the illegitimate updates. The evaluator performs this test using all of the following forms of illegitimate updates: 1) A modified version (e.g. using a hex editor) of a legitimately signed update If the TOE allows a delayed activation of updates the TOE must be able to display both the currently executing version and most recently installed version. The handling of version information of the most recently installed version might differ between different TOEs depending on the point in time when an attempted update is rejected. The evaluator shall verify that the TOE handles the most recently installed version information for that case as described in the guidance documentation. After the TOE has rejected the update the evaluator shall verify, that both, current version and most recently installed version, reflect the same version information as prior to the update attempt. The evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all methods supported (manual updates, automatic checking for updates, automatic updates).

	For distributed TOEs the evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all TOE components.
Test Steps	• Verify the current firmware version on the TOE. • Verify the current active and inactive images on TOE. • Using a Hex editor modify an otherwise good firmware image. • Upload the modified image on the TOE. • Attempt to install the modified image and verify that it fails. • Verify the image installation failure with logs. • Verify that the TOE firmware version has not changed. • Verify the current active and inactive images on TOE are not changed.
Expected Test Results	• The TOE should detect and reject the modified image for software update. • TOE logs should show image installation failure.
Pass/Fail with Explanation	Pass. The TOE software was able to detect when an image was corrupted and rejected the image. This meets the testing requirements.

7.6.4 FPT_TUD_EXT.1 TEST #2 (B)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 2 [conditional]: If the TOE itself verifies a digital signature to authorize the installation of an image to update the TOE the following test shall be performed (otherwise the test shall be omitted). The evaluator first confirms that no updates are pending and then performs the version verification activity to determine the current version of the product, verifying that it is different from the version claimed in the update(s) to be used in this test. The evaluator obtains or produces illegitimate updates as defined below and attempts to install them on the TOE. The evaluator verifies that the TOE rejects all of the illegitimate updates. The evaluator performs this test using all of the following forms of illegitimate updates: 2) An image that has not been signed

	If the TOE allows a delayed activation of updates the TOE must be able to display both the currently executing version and most recently installed version. The handling of version information of the most recently installed version might differ between different TOEs depending on the point in time when an attempted update is rejected. The evaluator shall verify that the TOE handles the most recently installed version information for that case as described in the guidance documentation. After the TOE has rejected the update the evaluator shall verify, that both, current version and most recently installed version, reflect the same version information as prior to the update attempt. The evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all methods supported (manual updates, automatic checking for updates, automatic updates). For distributed TOEs the evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all TOE components.
Test Steps	• Verify the current firmware version on the TOE. • Verify the current active and inactive images on TOE. • Modify the original image such that it does not have the signature. • Upload an image with no signature on the TOE. • Attempt to install the modified image and verify that it fails. • Verify the image installation failure with logs. • Verify that the TOE firmware version has not changed. • Verify the current active and inactive images on TOE are not changed.
Expected Test Results	• The TOE should detect and reject the image without signature for a software update. • TOE logs should show image installation failure.
Pass/Fail with Explanation	Pass. The TOE software was able to detect when an image was not signed and rejected the image. This meets the testing requirements.

7.6.5 FPT_TUD_EXT.1 TEST #2 (C)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 2 [conditional]: If the TOE itself verifies a digital signature to authorize the installation of an image to update the TOE the following test shall be performed (otherwise the test shall be omitted). The evaluator first confirms that no updates are pending and then performs the version verification activity to determine the

	current version of the product, verifying that it is different from the version claimed in the update(s) to be used in this test. The evaluator obtains or produces illegitimate updates as defined below and attempts to install them on the TOE. The evaluator verifies that the TOE rejects all of the illegitimate updates. The evaluator performs this test using all of the following forms of illegitimate updates: 3) An image signed with an invalid signature (e.g. by using a different key as expected for creating the signature or by manual modification of a legitimate signature) If the TOE allows a delayed activation of updates the TOE must be able to display both the currently executing version and most recently installed version. The handling of version information of the most recently installed version might differ between different TOEs depending on the point in time when an attempted update is rejected. The evaluator shall verify that the TOE handles the most recently installed version information for that case as described in the guidance documentation. After the TOE has rejected the update the evaluator shall verify, that both, current version and most recently installed version, reflect the same version information as prior to the update attempt. The evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all methods supported (manual updates, automatic checking for updates, automatic updates). For distributed TOEs the evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all TOE components.
Test Steps	• Verify the current firmware version on the TOE. • Verify the current active and inactive images on TOE. • Modify the original update image such that it has an invalid signature. • Upload the image with the invalid signature on the TOE. • Attempt to install the modified image and verify that it fails. • Verify the image installation failure with logs. • Verify that the TOE firmware version has not changed. • Verify the current active and inactive images on TOE are not changed.
Expected Test Results	• The TOE should detect and reject the image with an invalid signature for the software update. • TOE logs should show image installation failure.
Pass/Fail with Explanation	Pass. The TOE software was able to detect when an image had an invalid signature and rejected the image. This meets the testing requirements.

7.6.6 FPT_TUD_EXT.1 TEST #3 (A)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 3 [conditional]: If the TOE itself verifies a hash value over an image against a published hash value (i.e. reference value) that has been imported to the TOE from outside such that the TOE itself authorizes the installation of an image to update the TOE, the following test shall be performed (otherwise the test shall be omitted. If the published hash is provided to the TOE by the Security Administrator and the verification of the hash value over the update file(s) against the published hash is performed by the TOE, then the evaluator shall perform the following tests. The evaluator first confirms that no update is pending and then performs the version verification activity to determine the current version of the product, verifying that it is different from the version claimed in the update(s) to be used in this test. 1) The evaluator obtains or produces an illegitimate update such that the hash of the update does not match the published hash. The evaluator provides the published hash value to the TOE and calculates the hash of the update either on the TOE itself (if that functionality is provided by the TOE), or else outside the TOE. The evaluator confirms that the hash values are different, and attempts to install the update on the TOE, verifying that this fails because of the difference in hash values (and that the failure is logged). Depending on the implementation of the TOE, the TOE might not allow the Security Administrator to even attempt updating the TOE after the verification of the hash value fails. In that case the verification that the hash comparison fails is regarded as sufficient verification of the correct behaviour of the TOE If the TOE allows delayed activation of updates, the TOE must be able to display both the currently executing version and most recently installed version. The handling of version information of the most recently installed version might differ between different TOEs. Depending on the point in time when the attempted update is rejected, the most recently installed version might or might not be updated. The evaluator shall verify that the TOE handles the most recently installed version information for that case as described in the guidance documentation. After the TOE has rejected the update the evaluator shall verify,

	that both, current version and most recently installed version, reflect the same version information as prior to the update attempt. If the verification of the hash value over the update file(s) against the published hash is not performed by the TOE, Test 3 shall be skipped. The evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all methods supported (manual updates, automatic checking for updates, automatic updates). For distributed TOEs the evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all TOE components.
Pass/Fail with Explanation	N/A. The TOE does not support published hash verification.

7.6.7 FPT_TUD_EXT.1 TEST #3 (B)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 3 [conditional]: If the TOE itself verifies a hash value over an image against a published hash value (i.e. reference value) that has been imported to the TOE from outside such that the TOE itself authorizes the installation of an image to update the TOE, the following test shall be performed (otherwise the test shall be omitted. If the published hash is provided to the TOE by the Security Administrator and the verification of the hash value over the update file(s) against the published hash is performed by the TOE, then the evaluator shall perform the following tests. The evaluator first confirms that no update is pending and then performs the version verification activity to determine the current version of the product, verifying that it is different from the version claimed in the update(s) to be used in this test. 2) The evaluator uses a legitimate update and tries to perform verification of the hash value without providing the published hash value to the TOE. The evaluator confirms that this attempt fails. Depending on the implementation of the TOE it might not be possible to attempt the verification of the hash value without providing a hash value to the TOE, e.g. if the hash value needs to be handed over to the TOE as a parameter in a command line message and the syntax check of the command prevents the execution of the command without providing a hash value.

	In that case the mechanism that prevents the execution of this check shall be tested accordingly, e.g. that the syntax check rejects the command without providing a hash value, and the rejection of the attempt is regarded as sufficient verification of the correct behaviour of the TOE in failing to verify the hash. The evaluator then attempts to install the update on the TOE (in spite of the unsuccessful hash verification) and confirms that this fails. Depending on the implementation of the TOE, the TOE might not allow to even attempt updating the TOE after the verification of the hash value fails. In that case the verification that the hash comparison fails is regarded as sufficient verification of the correct behaviour of the TOE If the TOE allows delayed activation of updates, the TOE must be able to display both the currently executing version and most recently installed version. The handling of version information of the most recently installed version might differ between different TOEs. Depending on the point in time when the attempted update is rejected, the most recently installed version might or might not be updated. The evaluator shall verify that the TOE handles the most recently installed version information for that case as described in the guidance documentation. After the TOE has rejected the update the evaluator shall verify, that both, current version and most recently installed version, reflect the same version information as prior to the update attempt. If the verification of the hash value over the update file(s) against the published hash is not performed by the TOE, Test 3 shall be skipped. The evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all methods supported (manual updates, automatic checking for updates, automatic updates). For distributed TOEs the evaluator shall perform Test 1, Test 2 and Test 3 (if applicable) for all TOE components.
Pass/Fail with Explanation	N/A. The TOE does not support published hash verification.

7.7 X509-REV

7.7.1 FIA_X509_EXT.1.1/REV TEST #1A

Item	Data
Test Assurance Activity	Test 1a: The evaluator shall present the TOE with a valid chain of certificates (terminating in a trusted CA certificate) as needed to validate the leaf certificate to be used in the function and shall use this chain to demonstrate that the function succeeds. Test 1a shall be designed in a way that the chain can be 'broken' in Test 1b by either being able to remove the trust anchor from the TOEs trust store, or by setting up the trust store in a way that at least one intermediate CA certificate needs to be provided, together with the leaf certificate from outside the TOE, to complete the chain (e.g. by storing only the root CA certificate in the trust store).
Test Steps	• Configure TOE to connect to the TLS server. • Create a full chain of certificates to connect to the TOE. • Upload a complete certificate chain used for validation onto the TOE. • Establish a connection with the TOE over TLS and verify the successful connection. • Verify the successful connection with packet capture.
Expected Test Results	• When a complete certificate chain is present, the TOE should establish a successful TLS connection. • Packet capture should show a successful connection as a complete chain of certificates is present on the TOE.
Pass/Fail with Explanation	Pass. The TOE can make a successful connection when a complete certificate trust chain is present. This meets the test requirements.

7.7.2 FIA_X509_EXT.1.1/REV TEST #1B

Item	Data
Test Assurance Activity	Test 1b: The evaluator shall then 'break' the chain used in Test 1a by either removing the trust anchor in the TOE's trust store used to terminate the chain, or by removing one of the intermediate CA certificates (provided together with the leaf certificate in Test 1a) to complete the chain. The evaluator shall show that an attempt to validate this broken chain fails.
Test Steps	• Remove the ICA certificate from the TOE's trust store. • Establish a connection with the TOE over TLS and verify the connection. • Verify the failure logs on the device, showing the certificate verify failed. • Verify the unsuccessful connection with packet capture.

Expected Test Results	• The TOE should reject the connection when the ICA certificate is removed from the TOE's trust store. • TOE log should show certificate verification failure. • Packet capture should show connection failure as the intermediate CA certificate is removed from TOE.
Pass/Fail with Explanation	Pass. The TOE rejects the connection when an incomplete certificate trust chain is present. This meets the test requirements.

7.7.3 FIA_X509_EXT.1.1/REV TEST #2

Item	Data
Test Assurance Activity	The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. Test 2: The evaluator shall demonstrate that validating an expired certificate results in the function failing.
Test Steps	The ST states that "The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for all TLS and HTTPS peer entities and X.509 certificates are not used for either trusted updates or firmware integrity self-tests". • Create a server certificate that is expired. • Show the clock on the TOE. • Establish a connection with the TOE over TLS with an expired server certificate and verify that it fails. • Verify the failure logs on the device, showing connection is not established due to an expired certificate. • Verify the connection is unsuccessful via packet capture.
Expected Test Results	• The TOE should deny connection when the certificate is expired. • TOE logs should show connection failure due to an expired server certificate. • Packet capture should show connection failure as an expired server certificate is used.
Pass/Fail with Explanation	Pass. The TOE rejects the connection when an expired certificate is used. This meets the test requirements.

7.7.4 FIA_X509_EXT.1.1/REV TEST #3

Item	Data
Test Assurance Activity	The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. Test 3: The evaluator shall test that the TOE can properly handle revoked certificates-–conditional on whether CRL or OCSP is selected; if both are selected, then a test shall be performed for each method. The evaluator shall test revocation of the peer certificate and revocation of the peer intermediate CA certificate i.e. the intermediate CA certificate should be revoked by the root CA. The evaluator shall ensure that a valid certificate is used, and that the validation function succeeds. The evaluator then attempts the test with a certificate that has been revoked (for each method chosen in the selection) to ensure when the certificate is no longer valid that the validation function fails. Revocation checking is only applied to certificates that are not designated as trust anchors. Therefore, the revoked certificate(s) used for testing shall not be a trust anchor.
Test Steps	The ST states that "The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for all TLS and HTTPS peer entities and X.509 certificates are not used for either trusted updates or firmware integrity self-tests". The CRL is not selected in the ST. TOE Only supports revocation checking using OCSP. • Enable OCSP checking on the TOE. 1. Valid Certificate: • Create a server certificate and ICA certificate with the URI of the OCSP responder. Create a signer certificate with OCSP Signing enabled for ICA and server certificate. • Import the CA certificates on the TOE. • Verify that all certificates are valid. • Establish a connection with the TOE over TLS and verify that it is successful. • Verify with the OCSP responder that the certificates are valid.

	• Verify the successful connection logs on the TOE. • Verify the successful connection with packet capture. 2. Invalid End Entity Certificate: • Revoke the server certificate. • Verify that the database shows that the server certificate is revoked. • Establish a connection with the TOE over TLS and verify that it fails. • Verify with the OCSP responder that the certificate is revoked. • Verify the failure logs on the TOE showing validation failed due to a revoked certificate. • Verify the unsuccessful connection with packet capture. 3. Invalid Intermediate CA Certificate: • Revoke the intermediate certificate. • Verify that the database shows that the certificate is revoked. • Establish a connection with the TOE over TLS and verify that it fails. • Verify with the OCSP responder that the certificate is revoked. • Verify the failure logs on the TOE showing validation failed due to a revoked certificate. • Verify the unsuccessful connection with packet capture.
Expected Test Results	• The TOE should reject any TLS server connection when either the intermediate certificate or the server certificate has been revoked. • The OCSP connection should show that the certificates have been revoked. • The Packet capture is expected to depict the specific certificate that is revoked, and the logs should verify that the TOE denies connection by denoting that the certificate has been revoked.
Pass/Fail with Explanation	Pass. The CRL is not selected in the ST. TOE Only supports revocation checking using OCSP. It successfully connects when unrevoked certificates are used and rejects connections using revoked certificates. This meets the testing requirements.

7.7.5 FIA_X509_EXT.1.1/REV TEST #4

Item	Data
Test Assurance Activity	The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when

	performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. If OCSP is selected, the evaluator shall configure the OCSP server or use a man-in-the-middle tool to present a certificate that does not have the OCSP signing purpose and verify that validation of the OCSP response fails. If CRL is selected, the evaluator shall configure the CA to sign a CRL with a certificate that does not have the cRLsign key usage bit set and verify that validation of the CRL fails.
Test Steps	The ST states that "The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for all TLS and HTTPS peer entities and X.509 certificates are not used for either trusted updates or firmware integrity self-tests". The CRL is not selected in the ST. TOE Only supports revocation checking using OCSP. • Generate a certificate that does NOT have OCSP signing EKU. • Establish a connection with the TOE over TLS and verify that it fails. • Use a certificate that does NOT have OCSP signing EKU in the OCSP responder. • Verify validation of certificate failed as CA certificate doesn't have OCSP signing EKU via TOE logs. • Verify the unsuccessful TLS connection with the help of packet capture.
Expected Test Results	• The TOE should not establish a TLS server connection when the OCSP signing purpose is missing in the signer certificate, resulting in validation failure. • The packet capture should display a handshake failure due to the absence of OCSP Signing. • The logs should indicate that the connection was rejected by OCSP due to certificate verification failure.
Pass/Fail with Explanation	Pass. The CRL is not selected in the ST. TOE Only supports revocation checking using OCSP. The TOE rejects connections when the delegated signer certificate in OCSP is invalid and does not have OCSP-signer EKU. This meets the testing requirements.

7.7.6 FIA_X509_EXT.1.1/REV TEST #5

Item	Data

Test Assurance Activity	The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. The evaluator shall modify any byte in the first eight bytes of the certificate and demonstrate that the certificate fails to validate. (The certificate will fail to parse correctly.)
Test Steps	The ST states that "The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for all TLS and HTTPS peer entities and X.509 certificates are not used for either trusted updates or firmware integrity self-tests". • Start the server using the acumen-tlsc-v2.2e tool with a modified byte within the first 8 bytes of the certificate, the connection should fail. • Verify error logs are generated on the TOE as a certificate with modified bytes is presented. • Verify the unsuccessful connection with packet capture.
Expected Test Results	• TOE should reject the connection when the first 8 bytes of the certificate are modified. • TOE should generate error logs when a certificate with modified bytes is presented. • Packet capture should show connection failure due to a certificate with modified bytes being presented.
Pass/Fail with Explanation	Pass. The evaluator modified the first eight bytes of the certificate being presented by the server and ensured that the certificate fails to validate, and the TLS handshake fails. This meets the test requirements.

7.7.7 FIA_X509_EXT.1.1/REV TEST #6

Item	Data
Test Assurance Activity	The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. The evaluator shall modify any byte in the certificate signatureValue field (see RFC5280 Sec. 4.1.1.3), which is normally the last field in the certificate, and

	demonstrate that the certificate fails to validate. (The signature on the certificate will not validate.)
Test Steps	The ST states that "The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for all TLS and HTTPS peer entities and X.509 certificates are not used for either trusted updates or firmware integrity self-tests". • Start the server using the acumen-tlsc-v2.2e tool with a modified byte in the signatureValue field of the certificate. • Verify the error with logs on the device showing certificate signature failure. • Verify the unsuccessful connection with packet capture.
Expected Test Results	• TOE should reject the connection when the last byte of the certificate is modified. • TOE should generate error logs when a certificate with modified bytes is presented. • Packet capture should show connection failure due to a certificate with modified bytes being presented.
Pass/Fail with Explanation	Pass. The TOE rejects connections when the byte in the certificate signatureValue field is modified. This meets the test requirements.

7.7.8 FIA_X509_EXT.1.1/REV TEST #7

Item	Data
Test Assurance Activity	The evaluator shall demonstrate that checking the validity of a certificate is performed when a certificate is used in an authentication step or when performing trusted updates (if FPT_TUD_EXT.2 is selected). It is not sufficient to verify the status of a X.509 certificate only when it is loaded onto the TOE. Test 7: The evaluator shall modify any byte in the public key of the certificate and demonstrate that the certificate fails to validate. (The hash of the certificate will not validate.)
Test Steps	The ST states that "The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for all TLS and HTTPS peer entities and X.509 certificates are not used for either trusted updates or firmware integrity self-tests". • Start the server using the acumen-tlsc-v2.2e tool with the modified public key in the certificate. • Verify the error logs on the device showing certificate signature failure. • Verify the unsuccessful connection with packet capture.

Expected Test Results	• The TOE should reject connections when the public key of the certificate is modified. • TOE should generate error logs showing certificate signature failure. • Packet capture should show connection failure as the certificate with the modified public key is presented.
Pass/Fail with Explanation	Pass. The TOE rejects connections when any byte is the public key of the certificate is modified. This meets the test requirements.

7.7.9 FIA_X509_EXT.1.1/REV TEST #8A [TD0527]

Item	Data
Test Assurance Activity	**(Conditional on support for EC certificates as indicated in FCS_COP.1/SigGen)** **(Conditional on support for a minimum certificate path length of three certificates)** **(Conditional on TOE ability to process CA certificates presented in certificate message)** The test shall be designed in a way such that only the EC root certificate is designated as a trust anchor, and by setting up the trust store in a way that the EC Intermediate CA certificate needs to be provided, together with the leaf certificate, from outside the TOE to complete the chain (e.g. by storing only the EC root CA certificate in the trust store). The evaluator shall present the TOE with a valid chain of EC certificates (terminating in a trusted CA certificate), where the elliptic curve parameters are specified as a named curve. The evaluator shall confirm that the TOE validates the certificate chain.
Test Steps	• Create the EC root CA certificate. • Create the EC intermediate CA certificate. • Create the EC node certificate. • Configure the TOE for the root certificate as a trust anchor. • Concatenate the CA certificates. • Establish a connection with the TOE over TLS and verify that it is successful. • Verify the successful connection with packet capture.
Expected Test Results	• Connection using a trusted chain of the EC leaf certificate, EC intermediate certificate, and EC root certificate should be successful. • Packet capture should show a successful connection.

Pass/Fail with Explanation	Pass. The TOE makes a successful connection when the trusted chain of the EC leaf certificate, EC intermediate certificate and EC root certificate is used. This meets the test requirements.

7.7.10 FIA_X509_EXT.1.1/REV TEST #8B [TD0527]

Item	Data
Test Assurance Activity	**(Conditional on support for EC certificates as indicated in FCS_COP.1/SigGen)** **(Conditional on support for a minimum certificate path length of three certificates)** **(Conditional on TOE ability to process CA certificates presented in certificate message)** The test shall be designed in a way such that only the EC root certificate is designated as a trust anchor, and by setting up the trust store in a way that the EC Intermediate CA certificate needs to be provided, together with the leaf certificate, from outside the TOE to complete the chain (e.g. by storing only the EC root CA certificate in the trust store). The evaluator shall present the TOE with a chain of EC certificates (terminating in a trusted CA certificate), where the intermediate certificate in the certificate chain uses an explicit format version of the Elliptic Curve parameters in the public key information field, and is signed by the trusted EC root CA, but having no other changes. The evaluator shall confirm the TOE treats the certificate as invalid.
Test Steps	• In the second part of the test, modify the Intermediate certificate with an explicit format version of the Elliptic Curve parameters in the public key information field. • Concatenate the CA certificates. • Configure the TOE for the root certificate as a trust anchor. • Attempt the connection from the TOE to the TLS Server and verify that it fails. • Verify the failure logs on the device showing certificate validation failed. • Verify the unsuccessful connection via packet capture.
Expected Test Results	• When the public key information is modified in the intermediate certificate on the TLS server, TOE is unable to make a successful connection. • TOE should generate error logs showing certificate validation failed.

	• Packet capture showing connection failure as a modified intermediate certificate is presented.
Pass/Fail with Explanation	Pass. The TOE rejects the connection when the intermediate certificate in the certificate chain uses an explicit format version of the Elliptic Curve parameters in the public key information field. This meets the test requirements.

7.7.11 FIA_X509_EXT.1.1/REV TEST #8C [TD0527]

Item	Data
Test Assurance Activity	**(Conditional on support for EC certificates as indicated in FCS_COP.1/SigGen)** **(Conditional on support for a minimum certificate path length of three certificates)** The evaluator shall establish a subordinate CA certificate, where the elliptic curve parameters are specified as a named curve, that is signed by a trusted EC root CA. The evaluator shall attempt to load the certificate into the trust store and observe that it is accepted into the TOE's trust store. The evaluator shall then establish a subordinate CA certificate that uses an explicit format version of the elliptic curve parameters, and that is signed by a trusted EC root CA. The evaluator shall attempt to load the certificate into the trust store and observe that it is rejected, and not added to the TOE's trust store.
Test Steps	• In the third part of the test Intermediate certificate is modified with a named curve with an explicit format in the public key information field and is loaded on the TOE. • Attempt to add the modified Intermediate certificate on the TOE. • Attempt the connection from the TOE to the TLS Server and verify connection fails. • Verify error logs on the device showing the ICA certificate has an invalid public key. • Verify packet capture showing connection failure.
Expected Test Results	• TOE should reject the connection when the public key information is modified in the intermediate certificate on the TLS server. • TOE should generate error logs showing the ICA certificate has an invalid public key. • Packet capture should show connection failure.
Pass/Fail with Explanation	Pass. The TOE rejects the connection when the public key information is modified in the intermediate certificate. This meets the testing requirements.

7.7.12 FIA_X509_EXT.1.2/REV TEST #1

Item	Data
Test Assurance Activity	The tests described must be performed in conjunction with the other certificate services assurance activities, including the functions in FIA_X509_EXT.2.1/Rev. The tests for the extendedKeyUsage rules are performed in conjunction with the uses that require those rules. Where the TSS identifies any of the rules for extendedKeyUsage fields (in FIA_X509_EXT.1.1) that are not supported by the TOE (i.e. where the ST is therefore claiming that they are trivially satisfied) then the associated extendedKeyUsage rule testing may be omitted. The goal of the following tests is to verify that the TOE accepts a certificate as a CA certificate only if it has been marked as a CA certificate by using basicConstraints with the CA flag set to True (and implicitly tests that the TOE correctly parses the basicConstraints extension as part of X509v3 certificate chain validation). For each of the following tests the evaluator shall create a chain of at least three certificates: - a self-signed root CA certificate, - an intermediate CA certificate and - a leaf (node) certificate. The properties of the certificates in the chain are adjusted as described in each individual test below (and this modification shall be the only invalid aspect of the relevant certificate chain). Test 1: The evaluator shall ensure that at least one of the CAs in the chain does not contain the basicConstraints extension. The evaluator confirms that the TOE rejects such a certificate at one (or both) of the following points: *(i) as part of the validation of the leaf certificate belonging to this chain;* *(ii) when attempting to add a CA certificate without the basicConstraints extension to the TOE's trust store (i.e. when attempting to install the CA certificate as one which will be retrieved from the TOE itself when validating future certificate chains).*
Test Steps	• Create an ICA with no basicConstraint extension. • Upload ICA to TOE. • Verify that the TOE generates the warning message and does not add the certificate to the default-ca-list.

	• Verify the error in logs on the device showing the certificate rejected due to basic constraint failure.
Expected Test Results	• The TOE should reject certificates signed by CA that do not contain the BasicConstraints Extension. • TOE should generate error logs showing the certificate was rejected due to basic constraint failure.
Pass/Fail with Explanation	Pass. The TOE rejects certificates signed by a CA that do not contain the basicConstraints extension. This meets the test requirements.

7.7.13 FIA_X509_EXT.1.2/REV TEST #2

Item	Data
Test Assurance Activity	The tests described must be performed in conjunction with the other certificate services assurance activities, including the functions in FIA_X509_EXT.2.1/Rev. The tests for the extendedKeyUsage rules are performed in conjunction with the uses that require those rules. Where the TSS identifies any of the rules for extendedKeyUsage fields (in FIA_X509_EXT.1.1) that are not supported by the TOE (i.e. where the ST is therefore claiming that they are trivially satisfied) then the associated extendedKeyUsage rule testing may be omitted. The goal of the following tests it to verify that the TOE accepts only certificates that have been marked as CA certificates by using basicConstraints with the CA flag set to True (and implicitly that the TOE correctly parses the basicConstraints extension as part of X509v3 certificate chain validation). For each of the following tests the evaluator shall create a chain of at least three certificates: - a self-signed root CA certificate, - an intermediate CA certificate and - a leaf (node) certificate. The properties of the certificates in the chain are adjusted as described in each individual test below (and this modification shall be the only invalid aspect of the relevant certificate chain). Test 2: The evaluator shall ensure that at least one of the CA certificates in the chain has a basicConstraints extension in which the CA flag is set to FALSE. The

	evaluator confirms that the TOE rejects such a certificate at one (or both) of the following points: (i) As part of the validation of the leaf certificate belonging to this chain; (ii) When attempting to add a CA certificate with the CA flag set to FALSE to the TOE's trust store (i.e. when attempting to install the CA certificate as one which will be retrieved from the TOE itself when validating future certificate chains).
Test Steps	• Modify the ICA certificate with the flag in the basicConstraints extension set to FALSE using the x509-mod tool. • Establish a connection with the TOE over TLS using the modified ICA certificate and verify that it fails. • Verify that the connection is not established through logs. • Verify that the connection fails through packet capture.
Expected Test Results	• The TOE should reject certificates signed by ICA that have the CA flag set to FALSE. • TOE should generate error logs showing the certificate verification failed. • Packet capture should show that a fatal alert is generated.
Pass/Fail with Explanation	Pass. The TOE rejects certificates signed by a CA that has the CA flag in the basicConstraints extension set to FALSE. This meets the test requirements.

7.7.14 FIA_X509_EXT.2 TEST #1

Item	Data
Test Assurance Activity	The evaluator shall perform the following test for each trusted channel: The evaluator shall demonstrate that using a valid certificate that requires certificate validation checking to be performed in at least some part by communicating with a non-TOE IT entity. The evaluator shall then manipulate the environment so that the TOE is unable to verify the validity of the certificate and observe that the action selected in FIA_X509_EXT.2.2 is performed. If the selected action is administrator-configurable, then the evaluator shall follow the guidance documentation to determine that all supported administrator-configurable options behave in their documented manner.

Test Steps	TOE connects with the OCSP server to validate certs when attempting a TLS connection – Covered in FIA_X509_EXT.1.1/Rev Test #3 • Configure the certificates showing the OCSP distribution point. • Manipulate the Environment so that TOE is unable to validate the certificate from the OCSP server. • Attempt the connection from the TOE to the TLS server and show the connection being unsuccessful. • Verify the connection refused due to certificate verification failure via logs. • Verify the packet capture for handshake failure.
Expected Test Results	• The TOE will reject the connection when validation checking of the certificate is not available. • The packet capture will depict a handshake failure while the logs should show a failure in establishing a connection.
Pass/Fail with Explanation	Pass. The TOE rejects certificates that cannot be verified via OCSP when the responder is down. This meets the testing requirements.

7.7.15 FIA_X509_EXT.3 TEST #1

Item	Data
Test Assurance Activity	The evaluator shall use the guidance documentation to cause the TOE to generate a Certification Request. The evaluator shall capture the generated message and ensure that it conforms to the format specified. The evaluator shall confirm that the Certification Request provides the public key and other required information, including any necessary user-input information.

Test Steps	• From the TOE, generate a CSR (Key Size 3072).
	• Examine the CSR contents. Ensure the CSR contains the following fields.
	  ○ Common Name
	  ○ Organization
	  ○ Organizational Unit
	  ○ Country
	
	• From the TOE, generate a CSR (Key Size 2048).
	• Examine the CSR contents. Ensure the CSR contains the following fields.
	  ○ Common Name
	  ○ Organization
	  ○ Organizational Unit
	  ○ Country

Expected Test Results	• The TOE should generate CSR containing the required fields selected in the SFR. • Evidence – snapshot showing required fields are configured.
Pass/Fail with Explanation	Pass. The TOE is able to generate a CSR with all of the requisite information. This meets the testing requirements.

7.7.16 FIA_X509_EXT.3 TEST #2

Item	Data
Test Assurance Activity	The evaluator shall demonstrate that validating a response message to a Certification Request without a valid certification path results in the function failing. The evaluator shall then load a certificate or certificates as trusted CAs needed to validate the certificate response message and demonstrate that the function succeeds.
Test Steps	• Generate a CSR (Certificate Signing Request) on the TOE. • Generate a signed certificate based on the generated CSR from an external CA. • Ensure that the full trust chain for the signed CA is not present on the TOE i.e. only the CA certificate is added. • Attempt to load the signed certificate on the TOE. • Verify the error logs generated. • Add the intermediate certificate to the TOE certificate store to ensure that the TOE has a full certificate path. • Verify from the logs that the intermediate certificate is installed. • Verify that the TOE installs a CSR response with a full trust path. • Verify that the certificate is installed via logs.
Expected Test Results	• The TOE should not validate a signed CSR if the full trust chain is not present. When a full trust chain is present, the TOE should validate the signed CSR. • TOE should generate logs for certificate installation.
Pass/Fail with Explanation	Pass. The TOE only installs a CSR response signed by a CA with a full trust pat and does not validate a signed CSR if the full trust chain is not present. This meets the testing requirements.

7.8 CRYPTO

7.8.1 FCS_CKM.1 RSA

Item	Data
Test Assurance Activity	Note: The following tests require the developer to provide access to a test platform that provides the evaluator with tools that are typically not found on factory products. Generation of long-term cryptographic keys (i.e. keys that are not ephemeral keys/session keys) might be performed automatically (e.g. during initial start-up). Testing of key generation must cover not only administrator invoked key generation but also automated key generation (if supported). **Key Generation for FIPS PUB 186-4 RSA Schemes** The evaluator shall verify the implementation of RSA Key Generation by the TOE using the Key Generation test. This test verifies the ability of the TSF to correctly produce values for the key components including the public verification exponent *e*, the private prime factors *p* and *q*, the public modulus *n* and the calculation of the private signature exponent d. Key Pair generation specifies 5 ways (or methods) to generate the primes p and q. These include: a) Random Primes: • Provable primes • Probable primes b) Primes with Conditions: • Primes p1, p2, q1, q2, p and q shall all be provable primes • Primes p1, p2, q1, and q2 shall be provable primes and p and q shall be probable primes • Primes p1, p2, q1, q2, p and q shall all be probable primes To test the key generation method for the Random Provable primes method and for all the Primes with Conditions methods, the evaluator must seed the TSF key generation routine with sufficient data to deterministically generate the RSA key pair. This includes the random seed(s), the public exponent of the RSA key, and the desired key length. For each key length supported, the evaluator shall have the TSF generate 25 key pairs. The evaluator shall verify the correctness of the TSF's implementation by comparing values generated by the TSF with those generated from a known good implementation.

Pass/Fail with Explanation	Algorithm: RSA KeyGen Key size / Modulus: 2048, 3072 CAVP #: **A2624** Pass. Based on these findings, this assurance activity is considered satisfied.

7.8.2 FCS_CKM.1 ECC

Item	Data
Test Assurance Activity	Note: The following tests require the developer to provide access to a test platform that provides the evaluator with tools that are typically not found on factory products. Generation of long-term cryptographic keys (i.e. keys that are not ephemeral keys/session keys) might be performed automatically (e.g. during initial start-up). Testing of key generation must cover not only administrator invoked key generation but also automated key generation (if supported). **Key Generation for Elliptic Curve Cryptography (ECC)** *FIPS 186-4 ECC Key Generation Test* For each supported NIST curve, i.e., P-256, P-384 and P-521, the evaluator shall require the implementation under test (IUT) to generate 10 private/public key pairs. The private key shall be generated using an approved random bit generator (RBG). To determine correctness, the evaluator shall submit the generated key pairs to the public key verification (PKV) function of a known good implementation. *FIPS 186-4 Public Key Verification (PKV) Test* For each supported NIST curve, i.e., P-256, P-384 and P-521, the evaluator shall generate 10 private/public key pairs using the key generation function of a known good implementation and modify five of the public key values so that they are incorrect, leaving five values unchanged (i.e., correct). The evaluator shall obtain in response a set of 10 PASS/FAIL values.
Pass/Fail with Explanation	Algorithm: ECDSA KeyGen

	Curves: P-256, P-384, P-521 CAVP #: **A2624** Pass. Based on these findings, this assurance activity is considered satisfied.

7.8.3 FCS_CKM.1 FFC [**TD0580**]

Item	Data
Test Assurance Activity	Note: The following tests require the developer to provide access to a test platform that provides the evaluator with tools that are typically not found on factory products. Generation of long-term cryptographic keys (i.e. keys that are not ephemeral keys/session keys) might be performed automatically (e.g. during initial start-up). Testing of key generation must cover not only administrator invoked key generation but also automated key generation (if supported). **Key Generation for Finite-Field Cryptography (FFC)** The evaluator shall verify the implementation of the Parameters Generation and the Key Generation for FFC by the TOE using the Parameter Generation and Key Generation test. This test verifies the ability of the TSF to correctly produce values for the field prime p, the cryptographic prime q (dividing p-1), the cryptographic group generator g, and the calculation of the private key x and public key y. The Parameter generation specifies 2 ways (or methods) to generate the cryptographic prime q and the field prime p: • Primes q and p shall both be provable primes • Primes q and field prime p shall both be probable primes and two ways to generate the cryptographic group generator g: • Generator g constructed through a verifiable process • Generator g constructed through an unverifiable process. The Key generation specifies 2 ways to generate the private key x: • len(q) bit output of RBG where 1 <=x <= q-1 • len(q) + 64 bit output of RBG, followed by a mod q-1 operation and a +1 operation, where 1<= x<=q-1.

	The security strength of the RBG must be at least that of the security offered by the FFC parameter set. To test the cryptographic and field prime generation method for the provable primes method and/or the group generator g for a verifiable process, the evaluator must seed the TSF parameter generation routine with sufficient data to deterministically generate the parameter set. For each key length supported, the evaluator shall have the TSF generate 25 parameter sets and key pairs. The evaluator shall verify the correctness of the TSF's implementation by comparing values generated by the TSF with those generated from a known good implementation. Verification must also confirm • g != 0,1 • q divides p-1 • g^q mod p = 1 • g^x mod p = y for each FFC parameter set and key pair. **FFC Schemes using "safe-prime" groups** Testing for FFC Schemes using safe-prime groups is done as part of testing in CKM.2.1.
Pass/Fail with Explanation	**Key Generation for Finite-Field Cryptography (FFC)** Algorithm: **DSA KeyGen (FIPS186-4)** Capabilities: MODP-2048 CAVP #: **A2624** **FFC Schemes using "safe-prime" groups** Testing for FFC Schemes using safe-prime groups is done as part of testing in CKM.2.1. Pass. Based on these findings, this assurance activity is considered satisfied.

7.8.4 FCS_CKM.2 SP800-56A

Item	Data
Test Assurance Activity	**Key Establishment Schemes** The evaluator shall verify the implementation of the key establishment schemes of the supported by the TOE using the applicable tests below. **SP800-56A Key Establishment Schemes** The evaluator shall verify a TOE's implementation of SP800-56A key agreement schemes using the following Function and Validity tests. These validation tests for each key agreement scheme verify that a TOE has implemented the components of the key agreement scheme according to the specifications in the Recommendation. These components include the calculation of the DLC primitives (the shared secret value Z) and the calculation of the derived keying material (DKM) via the Key Derivation Function (KDF). If key confirmation is supported, the evaluator shall also verify that the components of key confirmation have been implemented correctly, using the test procedures described below. This includes the parsing of the DKM, the generation of MACdata and the calculation of MACtag. *Function Test* The Function test verifies the ability of the TOE to implement the key agreement schemes correctly. To conduct this test the evaluator shall generate or obtain test vectors from a known good implementation of the TOE supported schemes. For each supported key agreement scheme-key agreement role combination, KDF type, and, if supported, key confirmation role- key confirmation type combination, the tester shall generate 10 sets of test vectors. The data set consists of one set of domain parameter values (FFC) or the NIST approved curve (ECC) per 10 sets of public keys. These keys are static, ephemeral or both depending on the scheme being tested. The evaluator shall obtain the DKM, the corresponding TOE's public keys (static and/or ephemeral), the MAC tag(s), and any inputs used in the KDF, such as the Other Information field OI and TOE id fields. If the TOE does not use a KDF defined in SP 800-56A, the evaluator shall obtain only the public keys and the hashed value of the shared secret. The evaluator shall verify the correctness of the TSF's implementation of a given scheme by using a known good implementation to calculate the shared secret

	value, derive the keying material DKM, and compare hashes or MAC tags generated from these values. If key confirmation is supported, the TSF shall perform the above for each implemented approved MAC algorithm. *Validity Test* The Validity test verifies the ability of the TOE to recognize another party's valid and invalid key agreement results with or without key confirmation. To conduct this test, the evaluator shall obtain a list of the supporting cryptographic functions included in the SP800-56A key agreement implementation to determine which errors the TOE should be able to recognize. The evaluator generates a set of 24 (FFC) or 30 (ECC) test vectors consisting of data sets including domain parameter values or NIST approved curves, the evaluator's public keys, the TOE's public/private key pairs, MACTag, and any inputs used in the KDF, such as the other info and TOE id fields. The evaluator shall inject an error in some of the test vectors to test that the TOE recognizes invalid key agreement results caused by the following fields being incorrect: the shared secret value Z, the DKM, the other information field OI, the data to be MACed, or the generated MACTag. If the TOE contains the full or partial (only ECC) public key validation, the evaluator will also individually inject errors in both parties' static public keys, both parties' ephemeral public keys and the TOE's static private key to assure the TOE detects errors in the public key validation function and/or the partial key validation function (in ECC only). At least two of the test vectors shall remain unmodified and therefore should result in valid key agreement results (they should pass). The TOE shall use these modified test vectors to emulate the key agreement scheme using the corresponding parameters. The evaluator shall compare the TOE's results with the results using a known good implementation verifying that the TOE detects these errors.
Pass/Fail with Explanation	Algorithm: KAS-ECC-SSC Sp800-56Ar3 Curves: P-256, P-384, P-521 CAVP #: **A2624** Pass. Based on these findings, this assurance activity is considered satisfied.

7.8.5 FCS_CKM.2 RSA

Item	Data
Test Assurance Activity	**RSA-based key establishment** The evaluator shall verify the correctness of the TSF's implementation of RSAES-PKCS1-v1_5 by using a known good implementation for each protocol selected in FTP_TRP.1/Admin, FTP_TRP.1/Join, FTP_ITC.1 and FPT_ITT.1 that uses RSAES-PKCS1-v1_5.
Pass/Fail with Explanation	N/A. RSA-based key establishment is not claimed in ST.

7.8.6 FCS_CKM.2 FCC

Item	Data
Test Assurance Activity	**FFC Schemes using "safe-prime" groups** The evaluator shall verify the correctness of the TSF's implementation of safe-prime groups by using a known good implementation for each protocol selected in FTP_TRP.1/Admin, FTP_TRP.1/Join, FTP_ITC.1 and FPT_ITT.1 that uses safe-prime groups. This test must be performed for each safe-prime group that each protocol uses.
Pass/Fail with Explanation	Pass. This test has been successfully tested in FTP_TRP.1/Admin Test #1, FTP_ITC.1 Test#1 and FCS_SSHS_EXT.1.7 Test #2 since only SSH SFRs use safe-prime groups. The evaluator tested each protocol and verified the successful connection.

7.8.7 FCS_COP.1/DATAENCRYPTION AES-CBC KAT

Item	Data
Test Assurance Activity	**AES-CBC Known Answer Tests** There are four Known Answer Tests (KATs), described below. In all KATs, the plaintext, ciphertext, and IV values shall be 128-bit blocks. The results from each test may either be obtained by the evaluator directly or by supplying the inputs to the implementer and receiving the results in response. To determine correctness, the evaluator shall compare the resulting values to those obtained by submitting the same inputs to a known good implementation. **KAT-1.** To test the encrypt functionality of AES-CBC, the evaluator shall supply a set of 10 plaintext values and obtain the ciphertext value that results from AES-CBC encryption of the given plaintext using a key value of all zeros and an IV of all zeros. Five plaintext values shall be encrypted with a 128-bit all-zeros key, and the other five shall be encrypted with a 256-bit all-zeros key. To test the decrypt functionality of AES-CBC, the evaluator shall perform the same test as for encrypt, using 10 ciphertext values as input and AES-CBC decryption. **KAT-2.** To test the encrypt functionality of AES-CBC, the evaluator shall supply a set of 10 key values and obtain the ciphertext value that results from AES-CBC encryption of an all-zeros plaintext using the given key value and an IV of all zeros. Five of the keys shall be 128-bit keys, and the other five shall be 256-bit keys. To test the decrypt functionality of AES-CBC, the evaluator shall perform the same test as for encrypt, using an all-zero ciphertext value as input and AESCBC decryption. **KAT-3**. To test the encrypt functionality of AES-CBC, the evaluator shall supply the two sets of key values described below and obtain the ciphertext value that results from AES encryption of an all-zeros plaintext using the given key value and an IV of all zeros. The first set of keys shall have 128 128-bit keys, and the second set shall have 256 256-bit keys. Key i in each set shall have the leftmost i bits be ones and the rightmost N-i bits be zeros, for i in [1,N]. To test the decrypt functionality of AES-CBC, the evaluator shall supply the two sets of keys and ciphertext value pairs described below and obtain the plaintext value that results from AES-CBC decryption of the given ciphertext using the given key and an IV of all zeros. The first set of key/ciphertext pairs shall have 128 128-bit key/ciphertext pairs, and the second set of key/ciphertext pairs shall

	have 256 256-bit key/ciphertext pairs. Key i in each set shall have the leftmost i bits be ones and the rightmost N-i bits be zeros, for i in [1,N]. The ciphertext value in each pair shall be the value that results in an all-zeros plaintext when decrypted with its corresponding key. **KAT-4.** To test the encrypt functionality of AES-CBC, the evaluator shall supply the set of 128 plaintext values described below and obtain the two ciphertext values that result from AES-CBC encryption of the given plaintext using a 128-bit key value of all zeros with an IV of all zeros and using a 256- bit key value of all zeros with an IV of all zeros, respectively. Plaintext value i in each set shall have the leftmost i bits be ones and the rightmost 128-i bits be zeros, for i in [1,128]. To test the decrypt functionality of AES-CBC, the evaluator shall perform the same test as for encrypt, using ciphertext values of the same form as the plaintext in the encrypt test as input and AES-CBC decryption.
Pass/Fail with Explanation	Algorithm: AES CBC KAT Key size: 128, 256 CAVP #: **A2624** Pass. Based on these findings, this assurance activity is considered satisfied.

7.8.8 FCS_COP.1/DATAENCRYPTION AES-CBC MBMT

Item	Data
Test Assurance Activity	**AES-CBC Multi-Block Message Test** The evaluator shall test the encrypt functionality by encrypting an i-block message where 1 < i <=10. The evaluator shall choose a key, an IV and plaintext message of length i blocks and encrypt the message, using the mode to be tested, with the chosen key and IV. The ciphertext shall be compared to the result of encrypting the same plaintext message with the same key and IV using a known good implementation. The evaluator shall also test the decrypt functionality for each mode by decrypting an i-block message where 1 < i <=10. The evaluator shall choose a key, an IV and a ciphertext message of length i blocks and decrypt the message, using the mode to be tested, with the chosen key and IV. The plaintext shall be

	compared to the result of decrypting the same ciphertext message with the same key and IV using a known good implementation.
Pass/Fail with Explanation	Algorithm: AES CBC MBMT Key size: 128, 256 CAVP #: **A2624** Pass. Based on these findings, this assurance activity is considered satisfied.

7.8.9 FCS_COP.1/DATAENCRYPTION AES-CBC MCT

Item	Data

Test Assurance Activity	**AES-CBC Monte Carlo Tests**
	
	The evaluator shall test the encrypt functionality using a set of 200 plaintext, IV, and key 3-tuples. 100 of these shall use 128 bit keys, and 100 shall use 256 bit keys. The plaintext and IV values shall be 128-bit blocks. For each 3-tuple, 1000 iterations shall be run as follows:
	
	# Input: PT, IV, Key
	
	for i = 1 to 1000:
	
	if i == 1:
	
	CT[1] = AES-CBC-Encrypt(Key, IV, PT)
	
	PT = IV
	
	else:
	
	CT[i] = AES-CBC-Encrypt(Key, PT)
	
	PT = CT[i-1]
	
	The ciphertext computed in the 1000th iteration (i.e., CT[1000]) is the result for that trial. This result shall be compared to the result of running 1000 iterations with the same values using a known good implementation.

	The evaluator shall test the decrypt functionality using the same test as for encrypt, exchanging CT and PT and replacing AES-CBC-Encrypt with AESCBC-Decrypt.
Pass/Fail with Explanation	Algorithm: AES CBC MCT Key size: 128, 256 CAVP #: **A2624** Pass. Based on these findings, this assurance activity is considered satisfied.

7.8.10 FCS_COP.1/DATAENCRYPTION AES-GCM

Item	Data
Test Assurance Activity	**AES-GCM Test** The evaluator shall test the authenticated encrypt functionality of AES-GCM for each combination of the following input parameter lengths: **128 bit and 256 bit keys** a) **Two plaintext lengths.** One of the plaintext lengths shall be a nonzero integer multiple of 128 bits, if supported. The other plaintext length shall not be an integer multiple of 128 bits, if supported. a) **Three AAD lengths.** One AAD length shall be 0, if supported. One AAD length shall be a non-zero integer multiple of 128 bits, if supported. One AAD length shall not be an integer multiple of 128 bits, if supported. b) **Two IV lengths.** If 96 bit IV is supported, 96 bits shall be one of the two IV lengths tested. The evaluator shall test the encrypt functionality using a set of 10 key, plaintext, AAD, and IV tuples for each combination of parameter lengths above and obtain the ciphertext value and tag that results from AES-GCM authenticated encrypt. Each supported tag length shall be tested at least once per set of 10. The IV value may be supplied by the evaluator or the implementation being tested, as long as it is known. The evaluator shall test the decrypt functionality using a set of 10 key, ciphertext, tag, AAD, and IV 5-tuples for each combination of parameter lengths

	above and obtain a Pass/Fail result on authentication and the decrypted plaintext if Pass. The set shall include five tuples that Pass and five that Fail. The results from each test may either be obtained by the evaluator directly or by supplying the inputs to the implementer and receiving the results in response. To determine correctness, the evaluator shall compare the resulting values to those obtained by submitting the same inputs to a known good implementation.
Pass/Fail with Explanation	Algorithm: AES GCM Key size: 128, 256 CAVP #: **A2624** Pass. Based on these findings, this assurance activity is considered satisfied.

7.8.11 FCS_COP.1/DATAENCRYPTION AES-CTR KAT

Item	Data
Test Assurance Activity	**AES-CTR Known Answer Tests** The Counter (CTR) mode is a confidentiality mode that features the application of the forward cipher to a set of input blocks, called counters, to produce a sequence of output blocks that are exclusive-ORed with the plaintext to produce the ciphertext, and vice versa. Since the Counter Mode does not specify the counter that is used, it is not possible to implement an automated test for this mode. The generation and management of the counter is tested through FCS_SSH*_EXT.1.4. If CBC and/or GCM are selected in FCS_COP.1/DataEncryption, the test activities for those modes sufficiently demonstrate the correctness of the AES algorithm. If CTR is the only selection in FCS_COP.1/DataEncryption, the AES-CBC Known Answer Test, AESGCM Known Answer Test, or the following test shall be performed (all of these tests demonstrate the correctness of the AES algorithm): There are four Known Answer Tests (KATs) described below to test a basic AES encryption operation (AES-ECB mode). For all KATs, the plaintext, IV, and ciphertext values shall be 128-bit blocks. The results from each test may either be obtained by the validator directly or by supplying the inputs to the implementer and receiving the results in response. To determine correctness,

	the evaluator shall compare the resulting values to those obtained by submitting the same inputs to a known good implementation. KAT-1 To test the encrypt functionality, the evaluator shall supply a set of 5 plaintext values for each selected keysize and obtain the ciphertext value that results from encryption of the given plaintext using a key value of all zeros. KAT-2 To test the encrypt functionality, the evaluator shall supply a set of 5 key values for each selected keysize and obtain the ciphertext value that results from encryption of an all zeros plaintext using the given key value. KAT-3 To test the encrypt functionality, the evaluator shall supply a set of key values for each selected keysize as described below and obtain the ciphertext values that result from AES encryption of an all zeros plaintext using the given key values. A set of 128 128-bit keys, a set of 192 192-bit keys, and/or a set of 256 256-bit keys. Key i in each set shall have the leftmost i bits be ones and the rightmost N-i bits be zeros, for i in [1, N]. KAT-4 To test the encrypt functionality, the evaluator shall supply the set of 128 plaintext values described below and obtain the ciphertext values that result from encryption of the given plaintext using each selected keysize with a key value of all zeros (e.g. 256 ciphertext values will be generated if 128 bits and 256 bits are selected and 384 ciphertext values will be generated if all keysizes are selected). Plaintext value i in each set shall have the leftmost bits be ones and the rightmost 128-i bits be zeros, for i in [1, 128].
Pass/Fail with Explanation	Algorithm: AES CTR KAT Key size: 128, 256 CAVP #: **A2624** Pass. Based on these findings, this assurance activity is considered satisfied.

7.8.12 FCS_COP.1/DATAENCRYPTION AES-CTR MBMT

Item	Data
Test Assurance Activity	**AES-CTR Multi-Block Message Test**

	The evaluator shall test the encrypt functionality by encrypting an i-block message where 1 less-than i less-than-or-equal to 10 (test shall be performed using AES-ECB mode). For each i the evaluator shall choose a key and plaintext message of length i blocks and encrypt the message, using the mode to be tested, with the chosen key. The ciphertext shall be compared to the result of encrypting the same plaintext message with the same key using a known good implementation. The evaluator shall perform this test using each selected keysize.
Pass/Fail with Explanation	Algorithm: AES CTR MBMT Key size: 128, 256 CAVP #: **A2624** Pass. Based on these findings, this assurance activity is considered satisfied.

7.8.13 FCS_COP.1/DATAENCRYPTION AES-CTR MCT

Item	Data
Test Assurance Activity	**AES-CTR Monte-Carlo Test** The evaluator shall test the encrypt functionality using 100 plaintext/key pairs. The plaintext values shall be 128-bit blocks. For each pair, 1000 iterations shall be run as follows: # Input: PT, Key for i = 1 to 1000: CT[i] = AES-ECB-Encrypt(Key, PT) PT = CT[i] The ciphertext computed in the 1000th iteration is the result for that trial. This result shall be compared to the result of running 1000 iterations with the same values using a known good implementation. The evaluator shall perform this test using each selected keysize. There is no need to test the decryption engine.

Pass/Fail with Explanation	Algorithm: AES CTR MCT Key size: 128, 256 CAVP #: **A2624** Pass. Based on these findings, this assurance activity is considered satisfied.

7.8.14 FCS_COP.1/SIGGEN ECDSA

Item	Data
Test Assurance Activity	**ECDSA Algorithm Tests** **ECDSA FIPS 186-4 Signature Generation Test** For each supported NIST curve (i.e., P-256, P-384 and P-521) and SHA function pair, the evaluator shall generate 10 1024-bit long messages and obtain for each message a public key and the resulting signature values R and S. To determine correctness, the evaluator shall use the signature verification function of a known good implementation. **ECDSA FIPS 186-4 Signature Verification Test** For each supported NIST curve (i.e., P-256, P-384 and P-521) and SHA function pair, the evaluator shall generate a set of 10 1024-bit message, public key and signature tuples and modify one of the values (message, public key or signature) in five of the 10 tuples. The evaluator shall obtain in response a set of 10 PASS/FAIL values.
Pass/Fail with Explanation	Algorithm: ECDSA SigGen, SigVer Curves: P-256, P-384, P-521 CAVP #: **A2624** Pass. Based on these findings, this assurance activity is considered satisfied.

7.8.15 FCS_COP.1/SIGGEN RSA

Item	Data
Test Assurance Activity	**RSA Signature Algorithm Tests** **Signature Generation Test** The evaluator generates or obtains 10 messages for each modulus size/SHA combination supported by the TOE. The TOE generates and returns the corresponding signatures. The evaluator shall verify the correctness of the TOE's signature using a trusted reference implementation of the signature verification algorithm and the associated public keys to verify the signatures. **Signature Verification Test** For each modulus size/hash algorithm selected, the evaluator generates a modulus and three associated key pairs, (d, e). Each private key d is used to sign six pseudorandom messages each of 1024 bits using a trusted reference implementation of the signature generation algorithm. Some of the public keys, e, messages, or signatures are altered so that signature verification should fail. For both the set of original messages and the set of altered messages: the modulus, hash algorithm, public key e values, messages, and signatures are forwarded to the TOE, which then attempts to verify the signatures and returns the verification results. The evaluator verifies that the TOE confirms correct signatures on the original messages and detects the errors introduced in the altered messages.
Pass/Fail with Explanation	Algorithm: RSA SigGen, SigVer Key size / Modulus: 2048, 3072 CAVP #: **A2624** Pass. Based on these findings, this assurance activity is considered satisfied.

7.8.16 FCS_COP.1/HASH

Item	Data
Test Assurance Activity	The TSF hashing functions can be implemented in one of two modes. The first mode is the byte-oriented mode. In this mode the TSF only hashes messages that are an integral number of bytes in length; i.e., the length (in bits) of the message to be hashed is divisible by 8. The second mode is the bit-oriented mode. In this mode the TSF hashes messages of arbitrary length. As there are different tests for each mode, an indication is given in the following sections for the bit-oriented vs. the byte-oriented testmacs. The evaluator shall perform all of the following tests for each hash algorithm implemented by the TSF and used to satisfy the requirements of this PP. **Short Messages Test - Bit-oriented Mode** The evaluators devise an input set consisting of m+1 messages, where m is the block length of the hash algorithm. The length of the messages range sequentially from 0 to m bits. The message text shall be pseudorandomly generated. The evaluators compute the message digest for each of the messages and ensure that the correct result is produced when the messages are provided to the TSF. **Short Messages Test - Byte-oriented Mode** The evaluators devise an input set consisting of m/8+1 messages, where m is the block length of the hash algorithm. The length of the messages range sequentially from 0 to m/8 bytes, with each message being an integral number of bytes. The message text shall be pseudorandomly generated. The evaluators compute the message digest for each of the messages and ensure that the correct result is produced when the messages are provided to the TSF. **Selected Long Messages Test - Bit-oriented Mode** The evaluators devise an input set consisting of m messages, where m is the block length of the hash algorithm (e.g. 512 bits for SHA-256). The length of the ith message is m + 99*i, where 1 ≤ i ≤ m. The message text shall be pseudorandomly generated. The evaluators compute the message digest for each of the messages and ensure that the correct result is produced when the messages are provided to the TSF. **Selected Long Messages Test - Byte-oriented Mode** The evaluators devise an input set consisting of m/8 messages, where m is the block length of the hash algorithm (e.g. 512 bits for SHA-256). The length of the

	ith message is m + 8*99*i, where 1 ≤ i ≤ m/8. The message text shall be pseudorandomly generated. The evaluators compute the message digest for each of the messages and ensure that the correct result is produced when the messages are provided to the TSF. **Pseudorandomly Generated Messages Test** This test is for byte-oriented implementations only. The evaluators randomly generate a seed that is n bits long, where n is the length of the message digest produced by the hash function to be tested. The evaluators then formulate a set of 100 messages and associated digests by following the algorithm provided in Figure 1 of [SHAVS]. The evaluators then ensure that the correct result is produced when the messages are provided to the TSF.
Pass/Fail with Explanation	Algorithm: SHA-1, SHA-256, SHA-384, SHA-512 CAVP #: **A2624** Pass. Based on these findings, this assurance activity is considered satisfied.

7.8.17 FCS_COP.1/KEYEDHASH

Item	Data
Test Assurance Activity	For each of the supported parameter sets, the evaluator shall compose 15 sets of test data. Each set shall consist of a key and message data. The evaluator shall have the TSF generate HMAC tags for these sets of test data. The resulting MAC tags shall be compared to the result of generating HMAC tags with the same key and message data using a known good implementation.
Pass/Fail with Explanation	Algorithm: HMAC (SHA-1, SHA-256, SHA-384, SHA-512) CAVP #: **A2624** Pass. Based on these findings, this assurance activity is considered satisfied.

7.8.18 FCS_RBG_EXT.1

Item	Data
Test Assurance Activity	The evaluator shall perform 15 trials for the RNG implementation. If the RNG is configurable, the evaluator shall perform 15 trials for each configuration. If the RNG has prediction resistance enabled, each trial consists of (1) instantiate DRBG, (2) generate the first block of random bits (3) generate a second block of random bits (4) uninstantiate. The evaluator verifies that the second block of random bits is the expected value. The evaluator shall generate eight input values for each trial. The first is a count (0 – 14). The next three are entropy input, nonce, and personalization string for the instantiate operation. The next two are additional input and entropy input for the first call to generate. The final two are additional input and entropy input for the second call to generate. These values are randomly generated. “generate one block of random bits” means to generate random bits with number of returned bits equal to the Output Block Length (as defined in NIST SP800-90A). If the RNG does not have prediction resistance, each trial consists of (1) instantiate DRBG, (2) generate the first block of random bits (3) reseed, (4) generate a second block of random bits (5) uninstantiate. The evaluator verifies that the second block of random bits is the expected value. The evaluator shall generate eight input values for each trial. The first is a count (0 – 14). The next three are entropy input, nonce, and personalization string for the instantiate operation. The fifth value is additional input to the first call to generate. The sixth and seventh are additional input and entropy input to the call to reseed. The final value is additional input to the second generate call. The following paragraphs contain more information on some of the input values to be generated/selected by the evaluator. **Entropy input:** the length of the entropy input value must equal the seed length. **Nonce:** If a nonce is supported (CTR_DRBG with no Derivation Function does not use a nonce), the nonce bit length is one-half the seed length. **Personalization string:** The length of the personalization string must be <= seed length. If the implementation only supports one personalization string length, then the same length can be used for both values. If more than one string length is support, the evaluator shall use personalization strings of two different

	lengths. If the implementation does not use a personalization string, no value needs to be supplied. **Additional input:** the additional input bit lengths have the same defaults and restrictions as the personalization string lengths.
Pass/Fail with Explanation	Algorithm: Counter DRBG (AES-256), HMAC DRBG (SHA2-512) CAVP #: **A2624** Pass. Based on these findings, this assurance activity is considered satisfied.

8 CONCLUSION

The testing shows that all test cases required for conformance have passed testing.

9 REFERENCE

This section listed each piece of the evaluation evidence used by Acumen throughout the course of the evaluation. The evidence is listed in such a way as to uniquely identify each item.

- **FireEye AX, CM, EX, FX, HX, NX, and VX Series Appliances running TRFEOS 10.0.4 Security Target, version 2.0 (ST)**
- **FireEye AX, CM, EX, FX, HX, NX, and VX Series Appliances running TRFEOS 10.0.4 Guidance, version 1.4 (AGD)**