Assurance Activity Report for Ciena SAOS 10.7.1 on 3926, 3928, 3948, 5144, 5162, 5164, 5170, 5171, Large NFV Compute Server, and 8180 Service Aggregation Platforms

Ciena SAOS 10.7.1 on 3926, 3928, 3948, 5144, 5162, 5164, 5170, 5171, Large NFV Compute Server, and 8180 Service Aggregation Platforms Security Target

Version 1.3

collaborative Protection Profile for Network Devices Version 2.2e

AAR Version 0.8, March 11, 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:
Ciena Corporation
7035 Ridge RD,
Hanover, MD 21076

The Author of the Security Target:
Ciena Corporation

The TOE Evaluation was Sponsored by:
Ciena Corporation

Evaluation Personnel:
Kamran Farogh
Rupendra Kadtan
Joan Marshall
Fathi Nasraoui
Shaunak Shah

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
0.1	05/11/2023	Initial Release
0.2	12/10/2023	Updated the details
0.3	18/01/2024	QA Review
0.4	22/01/2024	QA review 2
0.5	09/02/2024	Separation of TSS and AGD sections
0.6	26/02/2024	Minor updates
0.7	05/03/2024	Addressing ECR comments
0.8	11/03/2024	Addressing ECR comments

Contents

1 TOE Overview

The TOE is the Ciena SAOS 10.7.1 on 3926, 3928, 3948, 5144, 5162, 5164, 5170, 5171, Large NFV Compute Server, and 8180 Service Aggregation Platforms. It is a non-distributed, non-virtual network device which implements routing and switching functionalities for enterprise, mobility, and converged network architectures. In these architectures, the TOE can be deployed in the access, aggregation, or core of the network. The TOE uses a Linux based container architecture for its SAOS Network Operating System and includes the Ciena SAOS 10.7.1 operating system executed on the Ciena 3926, 3928, 3948, 5144, 5162, 5164, 5170, 5171, Large NFV Compute Server, and 8180 Service Aggregation Platforms. The technical characteristics of the platforms are described in Sect.1.3.

The TOE implements the general functionality of a router/switch consistent with the collaborative Protection Profile for Network Devices v2.2E. The TOE implements controlled connectivity between two subnetworks and a management interface. All network traffic between the connected subnetworks is controlled by the TOE and the authorized administrators may manage the TOE using the management interface.

The management interface is a Command Line Interface (CLI) which may be accessed locally or remotely. Local access is via a console port which is a Serial EIA-561 (RJ-45) or a USB-C port. It allows management of the TOE from a workstation physically connected to the TOE. Remote management is over Secure Shell (SSH). SSH implements a secure remote login over a network connection and allows protected CLI.

All administrators are identified and authenticated using a username and password or based on SSH public key authentication. Access is only granted, and the user assigned to the role administrator upon successful authentication. Authentication is implemented locally. Authentication of TLS peers is done using X.509 Public Key Certificates. The validity of the X.509 public key certificates is verified using the Online Certificate Status Protocol (OCSP). TLS and Hypertext Transfer Protocol Security (HTTPS) may also be used for secure file transfer to and from the outside of the TOE.

In addition to the management ports for local and remote access by the administrators, the variants of the TOE also implement a different number of network ports for the interconnection of different subnetworks (see Sect.1.3). The network ports are physically separate from the management ports and administrative access may not take place from the network interconnection ports.

The TOE does not protect the data flowing through itself. The TOE is only to be deployed in a secure data center and to only be physically accessible by trusted administrators. Administrators are trusted to operate the TOE in accordance with the security guidance at all times and not attempt to circumvent or suppress the security functions and mechanisms of the TOE.

2 Assurance Activities Identification

The Assurance Activities contained within this document include all those defined within the NDcPP 2.2e based upon the mandatory and optional SFRs and those implemented based on selections within the PP.

2.1 References

In additional to TOE documentation, the following reference may also be valuable when understanding and controlling the TOE:

- Collaborative Protection Profile for Network Devices, Version 2.2e [NDcPP]
- AGD[1]: *Ciena SAOS 10.7.1 on 3926, 3928, 3948, 5144, 5162, 5164, 5170, 5171, Large NFV Compute Server, and 8180 Service Aggregation Platforms CC Guidance Supplement*.
- AGD[2]: *3948/513x/5144/516x/5170/811x Routers and Platforms Security SAOS 10.7.1*
- ST: *Ciena SAOS 10.7.1 on 3926, 3928, 3948, 5144, 5162, 5164, 5170, 5171, Large NFV Compute Server, and 8180 Service Aggregation Platforms Security Target*

3 Test Equivalency Justification

3.1 Architectural Description

The TOE is the Ciena SAOS 10.7.1 software executed on the Ciena 3926, 3928, 3948, 5144, 5162, 5164, 5170, 5171, Large NFV Compute Server, and 8180 Service Aggregation Platforms summarized in Table 1. The same software is executed on each platform. The various models of the TOE differ in performance and number of ports, but all run the same OS version 10.7.1 software. The TOE is available in two form factors:

1. a rack-mount appliance with a variable number of replaceable modules or 'blades', and
2. Large NFV Compute Server, a field-replaceable unit (FRU) housed in the 3926

Table 1: TOE Hardware Platforms

Models/Platform	1G/10G SFP+	Processors	100G	Power Options
3926	6	4x1.5GHz ARM Cortex A53	--	AC, DC
3928	4	4x1.5 GHz ARM Cortex A53	--	AC, DC
3948	4	4x1.5 GHz ARM Cortex A53	--	AC, DC
5144	8	4x2GHz ARM Cortex A72	--	AC, DC
5164	32x[1G/10G/25G]	4x2GHz ARM Cortex A72	4x [100G/ 200G]	AC, DC
5162	40	Intel XEON D1527, 4CORE	2	AC, DC

Models/Platform	1G/10G SFP+	Processors	100G	Power Options
5170	4x 25G/10G/1G and 36x 10G/1G	Intel XEON D1527, 4CORE	4xQSFP28	AC, DC
8180	--	Intel XEON D1527, 4CORE	32xQSFP28 FRU module options: 1xWLAi FRU and 4x100G CFP2-DCO	AC, DC
5171	4x 25G/10G/1G and 36x 10G/1G	Intel XEON D1539, 8CORE	FRU module options: 2x QSFP28, 1x QSFP28 + 1x 100G CFP2-DCO, 2x 100G CFP2-DCO, 1x200G CFP2-DCO	AC, DC
Large NFV compute server (FRU)	--	Intel XEON D1548, 8CORE	--	--

The TOE is deployed in an environment that includes the IT components illustrated in figure 1. The TOE itself is delivered as an appliance or an FRU with the software installed. The administrator of the TOE may verify the TOE software and, if necessary, download and install the correct version.

The physical boundary of the TOE is illustrated below. Non-TOE components are summarized in Table 2. The TOE relies on an external NTP server for precise time synchronization, employing the SHA-1 hash algorithm to secure NTP time stamps against tampering. Communication with the NTP server is established through the UDP protocol, and authentication is ensured through the application of the SHA-1 hash algorithm. Additionally, the TOE interacts with a remote file server via HTTPS/TLS, facilitating the secure storage of user files. For certificate revocation status determination, the TOE integrates an OCSP responder within its operational environment, communicating with it through the HTTP protocol over the TCP protocol. To manage the TOE, an administrator employs a designated Management Workstation, capable of both local and remote management and equipped with an SSH client to facilitate secure communication.

Figure 1: TOE Boundary and Operational Environment

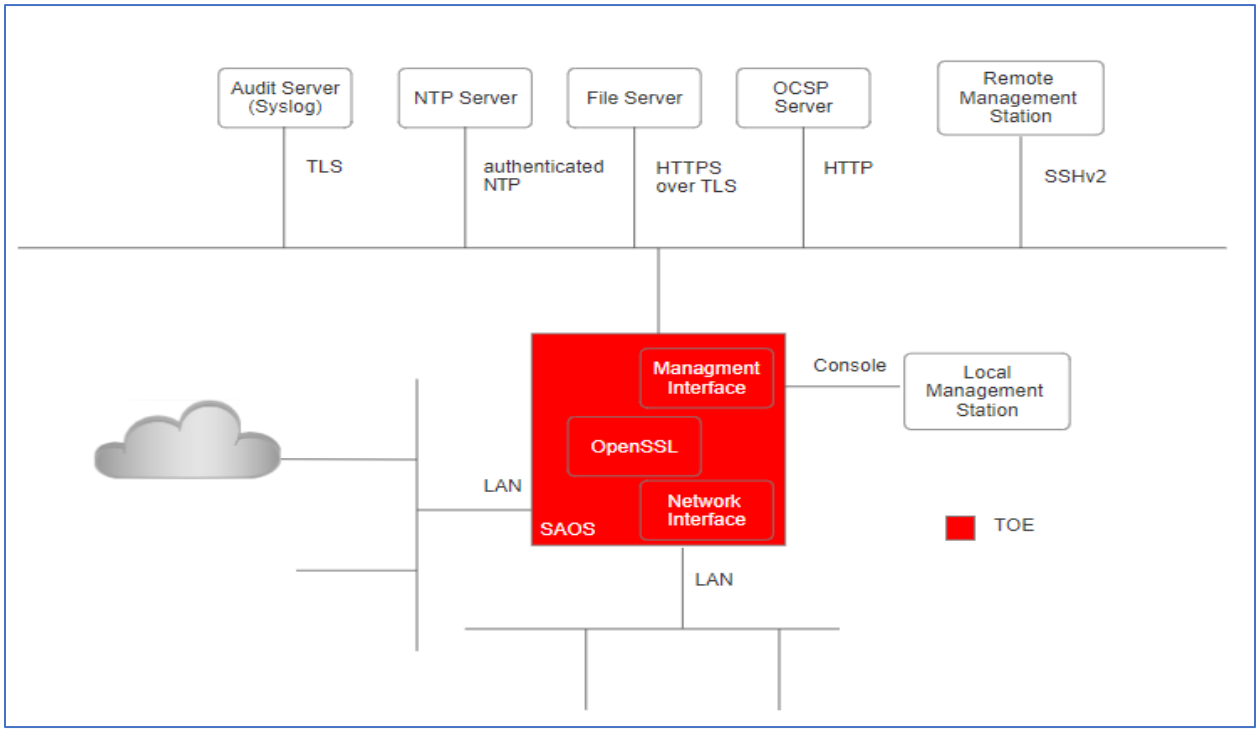


Table 2: TOE Operating Environment Component

Component	Purpose/Description
Audit server	The audit server supports syslog messages over TLS to receive the audit files from the TOE. The audit data is stored in the remote audit server for redundancy purposes.
NTP server	An external NTP Server for synchronizing the TOE time with. NTP time stamps are protected from tampering using SHA-1 for authentication.
File Server	Remote file server for storing user files and updating the TOE. Communication with the File Server is with HTTPS over TLS.
OCSP Server	Validity of the certificates the TOE uses for asserting the authenticity of the TLS peers is verified using OCSP. Communication with an OCSP Server is over HTTP.
Management Workstation	A workstation used by an administrator to manage the TOE locally or remotely. The remote management station must include a SSHv2 client.

3.2 Equivalency Analysis

The following equivalency analysis provides a per category analysis of key areas of differentiation for each hardware model to determine the minimum subset to be used in testing. The areas examined will use the areas and analysis description provided in the supporting documentation for the NDcPP V2.2e.

3.2.1 Platform/Hardware Dependencies

The TOE boundary is the hardware appliance, which is comprised of hardware and software components. It is deployed in an environment that contains various IT components.

The SFR enforcing code is identical and agnostic across all the devices drivers of the underlying device for the network and physical medium that is carried across. Also, all security functionality is implemented in Platform Independent code which is line-by-line identical across equivalent hardware models. The hardware within the TOE only differs by configuration and performance.

There are two separate CPU microarchitectures (ARMV8-A and Broadwell) used in the Ciena models.

Result: Each of the CPU microarchitectures should be tested.

3.2.2 Software/OS Dependencies:

This category of differences is only applicable if the TOE is installed on an OS outside of the TOE boundary. The OS for all models within the TOE is the Ciena SAOS 10.7.1. There are no specific dependencies on the OS since the TOE will not be installed on different OSs.

Result: All platforms are equivalent. There are no OS dependencies.

3.2.3 Differences in Libraries Used to Provide TOE Functionality

All software binaries compiled in the TOE software are identical and have the same version numbers. There are no differences between the included libraries. The cryptographic library included in SAOS has been CAVP certified on each of four different CPUs (ARM Cortex A53, ARM Cortex A72, Intel XEON D1527, and Intel XEON D1539 and can be viewed at A3495. The following table includes the Intel XEON D1548 is claiming CAVP equivalent to the Broadwell microarchitecture certified models and therefore, does not need a separate CAVP cert.

Result: All libraries are equivalent

3.2.4 Differences in Cryptographic Module Validation Claims

The Table below provides a summary of fully tested processors and their respective CAVP certificate numbers.

TOE Model	Operating Environment	CAVP Certificate #
3926	ARM Cortex A53 (ARMv8-A)	A3495 (fully tested)
3928	ARM Cortex A53 (ARMv8-A)	A3495

3948	ARM Cortex A53 (ARMv8-A)	A3495
5144	ARM Cortex A72 (ARMv8-A)	A3495
5164	ARM Cortex A72 (ARMv8-A)	A3495
5162	Intel XEON D1527 (Broadwell)	A3495 (fully tested)
5170	Intel XEON D1527 (Broadwell)	A3495
8180	Intel XEON D1527 (Broadwell)	A3495
5171	Intel XEON D1539 (Broadwell)	A3495
Large NFV Compute Server	Intel XEON D1548 (Broadwell)	Claimed Equivalent to A3495.

3.2.5 TOE Management Interface Differences

The TOE is managed via either remote CLI session or directly connected CLI. These management options are available on all hardware platforms regardless of the configuration. There is no difference in the management interface for any platform.

Result: All platforms are equivalent

3.2.6 TOE Functional Differences

Each hardware model within the TOE boundary provides identical functionality. There is no difference in the way the user interacts with each of the devices or the services that are available to the user in each of these devices. Each device runs the same version of SAOS software. For SAOS software, differences in the provided functionality are denoted by a different version of the software. If there had been differences in the functionality provided by the software, the actual release version would have been different for the platform.

Result: All platforms are equivalent.

3.2.7 Difference Comparison

The following table provides a comparison of each of the categories with differences.

(Device Group: The models who are sharing the same microarchitecture grouped together.)

The following table compares the ten appliances included in the TOE. The Test Group identifies each appliance into one of two Test Groups for CC Testing based on microarchitecture.

Table 3: TOE Appliances

TOE Model	Operating System	CPU	Microarchitecture	Test Group
3926	Ciena SAOS 10.7.1	4x1.5GHz ARM Cortex A53	ARMv8-A	1
3928	Ciena SAOS 10.7.1	4x1.5 GHz ARM Cortex A53	ARMv8-A	1
3948	Ciena SAOS 10.7.1	4x1.5 GHz ARM Cortex A53	ARMv8-A	1
5144	Ciena SAOS 10.7.1	4x2GHz ARM Cortex A72	ARMv8-A	1
5164	Ciena SAOS 10.7.1	4x2GHz ARM Cortex A72	ARMv8-A	1
5162	Ciena SAOS 10.7.1	Intel XEON D1527, 4CORE	Broadwell	2
5170	Ciena SAOS 10.7.1	Intel XEON D1527, 4CORE	Broadwell	2
8180	Ciena SAOS 10.7.1	Intel XEON D1527, 4CORE	Broadwell	2
5171	Ciena SAOS 10.7.1	Intel XEON D1539, 8CORE	Broadwell	2
Large NFV Compute Server	Ciena SAOS 10.7.1	Intel XEON D1548, 4 CORE	Broadwell	2

As the table above shows, all models are running the same version of the Ciena SAOS 10.7.1.

There are five main processors which are used by Ciena models, which are Intel series and ARM Cortex Series. The ARM Cortex A53 and ARM Cortex A72 share the same microarchitecture - ARMV8-A and the Intel XEON D1527, Intel XEON D1539, and Intel XEON D1548 share the same microarchitecture – Broadwell.

3.3 Recommendations/Conclusions

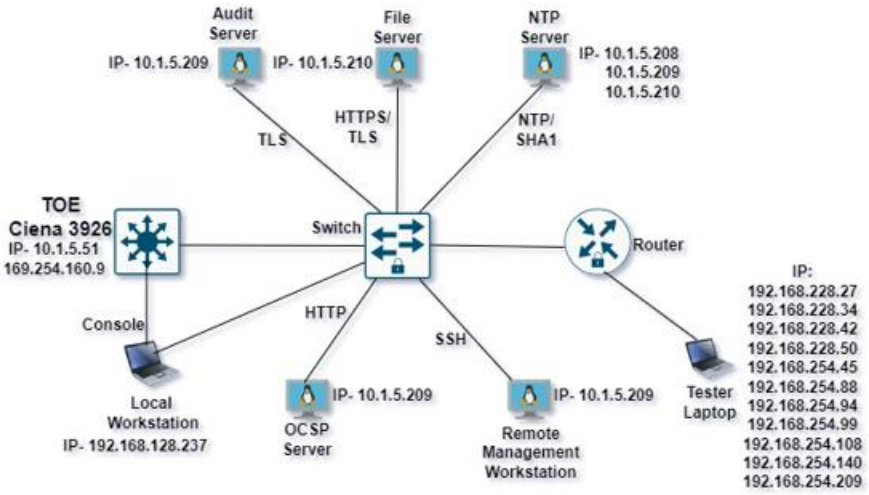
Based on the equivalency rationale listed above, testing will be performed on the following systems.

- Ciena 3926
- Ciena 5162

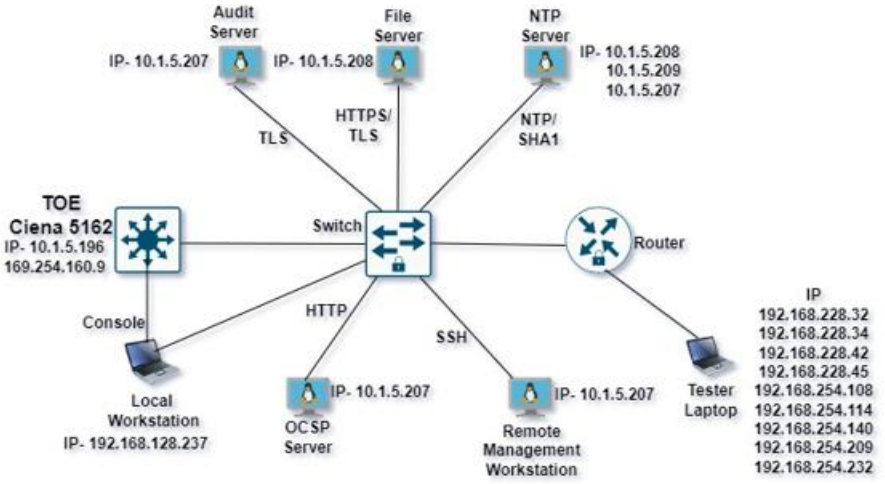
Model	Micro-Architecture	Vendor Name
3926	ARMV8-A	ARM Holdings
5162	Broadwell	Intel

4 Test Bed Descriptions

4.1 Test Beds - Audit, Auth, SSHS, TLSC, Update, X509-Rev (Ciena 3926)



4.2 Test Beds - Audit, Auth, SSHS, TLSC, Update, X509-Rev (Ciena 5162)



4.3 Test Bed Description (Ciena 3926)

Name	OS	Version	Function	Protocols	IP address	MAC Address	Time	Tools (version)
Ciena 3926	SAOS	10.7.1	TOE	SSH, TLS, HTTP, HTTPS, NTP	10.1.5.51	18:92:a4:03:e0:00	Manually set and verified	
Test VM1	Kali Linux	5.18.0-kali5-amd64	TOE Administration/ NTP Server	SSH, NTP	10.1.5.208	00:0c:29:26:2a:59	Manually set and verified	tcpdump version 4.99.1 libpcap version 1.10.1 (with TPACKET_V3) OpenSSH_8.7p1
Test VM2	Kali Linux	5.18.0-kali5-amd64	TOE Administration/ Syslog server/ Certificate authority/OCSP server/NTP Server	SSH, TLS, HTTP, NTP	10.1.5.209	00:0c:29:bf:90:1e	Manually set and verified	tcpdump version 4.99.1 libpcap version 1.10.1 (with TPACKET_V3) OpenSSL 1.1.1m 14 Dec 2021 OpenSSH_8.7p1
Test VM3	Kali Linux	5.18.0-kali5-amd64	TOE Administration/ File Server/ NTP server/	SSH, HTTPS, NTP	10.1.5.210	00:0c:29: 68:3e:3e	Manually set and verified	tcpdump version 4.99.1

Name	OS	Version	Function	Protocols	IP address	MAC Address	Time	Tools (version)
								libpcap version 1.10.1 (with TPACKET_V3) OpenSSL 1.1.1m 14 Dec 2021 OpenSSH_8.7p1
Tester Laptop	Microsoft	Windows 10 Pro	Test workstation	SSH, RDP	192.168.228.27 192.168.228.34 192.168.228.42 192.168.228.50 192.168.254.45 192.168.254.88 192.168.254.94 192.168.254.99 192.168.254.108 192.168.254.140 192.168.254.209	54-14-F3-E8-C4-2A	Manually set and verified	Mobaxterm v23.0 XCA 2.4.1, WinSCP v5.19.6 Wireshark v3.4.8 Putty v0.76 Hex editor 2.5
Local Management Station	Microsoft	Windows 10 Pro	Console workstation	SSH, RDP	192.168.128.237	9c-DA-3E-A8-F5-2A	Manually set and verified	Putty 0.76 Mobaxterm v23.0 WinSCP v5.19.6 Wireshark v3.4.8

4.4 Test Bed Description (Ciena 5162)

Name	OS	Version	Function	Protocols	IP address	MAC Address	Time	Tools (version)
Ciena 5162	SAOS	10.7.1	TOE	SSH, TLS, HTTP, HTTPS, NTP	10.1.5.196	14:4e:2a:0e:a2:80	Manually set and verified	
Test VM1	Kali Linux	5.18.0-kali5-amd64	TOE Administration/ Syslog server/ Certificate authority/OCSP server/ NTP Server	SSH, TLS, HTTP, UDP, NTP	10.1.5.207	00:0c:29:ff:f5:d1	Manually set and verified	tcpdump version 4.99.1 libpcap version 1.10.1 (with TPACKET_V3) OpenSSL 1.1.1f 31 Mar 2020 OpenSSH_9.0p1 Debian-1+b1 Ryslogd 8.2112.0
Test VM2	Kali Linux	5.18.0-kali5-amd64	TOE Administration/ File Server/NTP Server	SSH, HTTPS, NTP,	10.1.5.208	00:0c:29:26:2a:59	Manually set and verified	tcpdump version 4.99.1

Name	OS	Version	Function	Protocols	IP address	MAC Address	Time	Tools (version)
								libpcap version 1.10.1 (with TPACKET_V3) OpenSSL 3.0.7 1 Nov 2022 OpenSSH_9.1p1 Debian-1
Test VM3	Kali Linux	5.18.0-kali5-amd64	TOE Administration /NTP Server	SSH, NTP	10.1.5.209	00:0c:29:bf:90:1e	Manually set and verified	OpenSSL 3.0.7 1 Nov 2022 tcpdump version 4.99.1 libpcap version 1.10.1 (with TPACKET_V3)
Tester Laptop	Microsoft	Windows 10 Pro	Test workstation	SSH, RDP	192.168.228.32 192.168.228.34 192.168.228.42 192.168.228.45 192.168.254.108 192.168.254.114 192.168.254.140	54-14-F3-E8-C4-2A	Manually set and verified	Mobaxterm v23.0 XCA 2.4.1, WinSCP v5.19.6 Wireshark v3.4.8 Putty v0.76 Hex editor 2.5

Name	OS	Version	Function	Protocols	IP address	MAC Address	Time	Tools (version)
					192.168.254.209 192.168.254.232			
Local Management Station	Microsoft	Windows 10 Pro	Console workstation	SSH, RDP	192.168.128.237	9c-DA-3E-A8-F5-2A	Manually set and verified	Putty v0.76 Mobaxterm v23.0 WinSCP v5.19.6 Wireshark 3.4.8

4.5 Test Time and Location

All testing was conducted at the Acumen Security offices, situated at 2400 Research Blvd, Suite #395, Rockville, MD 20850. The testing spanned from April 2022 through January 2024.

The TOE was located in a physically protected, access-controlled, designated test lab with no unattended entry/exit points. At the beginning of each day, the test bed underwent verification to ensure its integrity. All evaluation documentation was consistently stored with the evaluator.

Regression testing was conducted in January 2024 on build 10.7.1_0289_RS12 for bug fixing and below mentioned test cases were tested during the same.

- FAU_STG_EXT.1 Test #1
- FCS_NTP_EXT1.1_T1
- FMT_MTD.1/CryptoKeys Test #1

- FMT_MTD.1/CryptoKeys Test #2
- FCS_SSHS_EXT.1.2 Test #1
- FCS_SSHS_EXT.1.2 Test #2
- FCS_SSHS_EXT.1.4 Test #1
- FCS_SSHS_EXT.1.5 Test #1
- FCS_SSHS_EXT.1.5 Test #2
- FCS_TLSC_EXT_1.1 Test 1
- FCS_TLSC_EXT_1.1 Test 4b
- FPT_TUD_EXT.1 Test #1
- FPT_TST_EXT.1.1 Test #1
- FPT_TUD_EXT.1 Test #2 (a)
- FPT_TUD_EXT.1 Test #2 (b)
- FPT_TUD_EXT.1 Test #2 (c)
- FIA_X509_EXT.1.1/Rev Test #1a
- FIA_X509_EXT.1.1/Rev Test #1b

5 Detailed TSS Assurance Activities

5.1 TSS Activities (Auditing)

5.1.1 FAU_GEN.1

5.1.1.1 **FAU_GEN.1 TSS 1**

Objective	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 section titled **TOE Summary Specifications** in the Security Target to determine the verdict of this assurance activity. The evaluator confirmed that within this section it identified the following information that was logged in order to identify the relevant key in relation to import/generation, changing, or deletion of cryptographic keys: **An audit log uniquely identifies a cryptographic key by its name, the operation performed on the relevant key, and the user identity performing the operation.**

	Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.1.2 FAU_STG.1

5.1.2.1 **FAU_STG.1 TSS 1**

Objective	The evaluator shall examine the TSS to ensure it describes the amount of audit data that are stored locally and how these records are protected against unauthorized modification or deletion. The evaluator shall ensure that the TSS describes the conditions that must be met for authorized deletion of audit records.
Evaluator Findings	The evaluator examined the section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS describes the amount of audit data that are stored locally, how these records are protected against unauthorized modification or deletion, and the conditions that must be met for authorized deletion of audit records. Upon investigation, the evaluator found that the TSS states that: **Audit records are stored persistently on the local file system. The TOE is a standalone component that stores audit data locally. The amount of audit data that may be stored locally on the TOE is dependent on the available disk space which varies depending on platform. When the local audit data store is exhausted, the TOE will overwrite audit records starting with the oldest audit record.** **Only authorized administrators may view and clear audit records using the CLI which is the sole interface to the management functions of the TOE.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.1.3 FAU_STG_EXT.1

5.1.3.1 **FAU_STG_EXT.1 TSS 1**

Objective	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 section 6.1 titled **TOE Summary Specifications** in the Security Target to verify that the TSS describes the means by which the audit data are transferred to the external audit server, and how the trusted channel is provided. Upon investigation, the evaluator found that the TSS states that: **The Security Administrator can configure the TOE to transfer the audit data to an external audit server. The audit logs are sent to the external audit server via TLS in real-time.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.1.3.2 FAU_STG_EXT.1 TSS 2

Objective	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 section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS 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. Upon investigation, the evaluator found that the TSS states that: **The TOE is a standalone component that stores audit data locally. The amount of audit data that may be stored locally on the TOE is dependent on the available disk space which varies depending on platform. When the local audit data store is exhausted, the TOE will overwrite audit records starting with the oldest audit record.** **Only authorized administrators may view and clear audit records using the CLI which is the sole interface to the management functions of the TOE.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.1.3.3 FAU_STG_EXT.1 TSS 3

Objective	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. The evaluator shall examine the TSS to ensure that for distributed TOEs it contains a list of TOE components that store audit data locally. 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 evaluator examined section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS 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. Upon investigation, the evaluator found that the TSS states that: **The TOE is a standalone component that stores audit data locally.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.1.3.4 FAU_STG_EXT.1 TSS 4

Objective	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 section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS details the behavior of the TOE when the storage space for audit data is full. Upon investigation, the evaluator found that the TSS states that: **The amount of audit data that may be stored locally on the TOE is dependent on the available disk space which varies depending on platform. When the local audit data store is exhausted, the TOE will overwrite audit records starting with the oldest audit record.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.1.3.5 FAU_STG_EXT.1 TSS 5

Objective	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 realtime or periodically. In case the TOE does not perform transmission in realtime 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 section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS details whether the transmission of audit information to an external IT entity can be done in realtime or periodically. Upon investigation, the evaluator found that the TSS states that: **The audit logs are sent to the external audit server via TLS in real-time.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.2 TSS Activities (Cryptographic Support)

Note that Test activities in the SD that are typically addressed by referencing CAVP certs are addressed in this Test section and are identified as "Test/CAVP" activities.

5.2.1 FCS_CKM.1

5.2.1.1 **FCS_CKM.1 TSS 1**

Objective	The evaluator shall ensure that the TSS identifies the key sizes supported by the TOE. 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 section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS identifies the key sizes supported by the TOE. Upon investigation, the evaluator found that the TSS states that: **The following table lists the key generation algorithms that are supported by the TOE along with the trusted communications protocols that they are used for:**

Algorithm/Protocol	TLS	SSHS
RSA	2048 bits, 4096 bits	2048 bits, 4096 bits
ECC	secp256r1, secp384r1, secp521r1	nistp256, nistp384, nistp521
FFC (DH Groups)	N/A	Group14, group16

	Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.2.2 FCS_CKM.2

5.2.2.1 FCS_CKM.2 TSS 1 [TD0580]

Objective	The evaluator shall ensure that the supported key establishment schemes correspond to the key generation schemes identified in FCS_CKM.1.1. 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.
Evaluator Findings	The evaluator examined section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS supported key establishment schemes correspond to the key generation schemes identified in FCS_CKM.1.1. Upon investigation, the evaluator found that the TSS states that: **The following table lists the key establishment schemes that are supported by the TOE along with the trusted communications protocols that they are used for:** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

	TLS (FCS_TLSC_EXT.1.1, FCS_TLSC_EXT.1.4)	SSH (FCS_SSHS_EXT.1.7)
RSA	2048 bits, 4096 bits	N/A
ECC	secp256r1, secp384r1, secp521r1	nistp256, nistp384, nistp521
FFC (DH Groups)	N/A	Group 14, Group 16

5.2.3 FCS_CKM.4

5.2.3.1 FCS_CKM.4 TSS 1

Objective	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 for2). 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 section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS 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. Upon investigation, the evaluator found that the TSS states that: **The TOE destroys cryptographic keys in accordance with a specified cryptographic key destruction method:** • **For plaintext keys in volatile storage, the destruction shall be executed by a single overwrite consisting of zeroes.** • **For plaintext keys in non-volatile storage, the TOE destroys keys by the SAOS overwriting the storage location of the key with a single overwrite of zeroes.** The evaluator examined the section titled **TOE Summary Specifications** in the Security Target to verify that the TSS description of keys and storage locations is consistent with the functions carried out by the TOE. Upon investigation, the evaluator found that the TSS states that: **The destruction of each cryptographic key and Critical Security Parameter (CSP) is summarized in Table 15.**

Table 4: Destruction of Key and CSP

Keys/CSPs	Purpose	Storage Location	Method of Zeroization
SSH Server Host Keys	The SSH server host keys to identify ssh server	Non-volatile storage/file system	Overwrite with zeros to clear cache and read verify, then erase file

	SSH session keys	Keys exchanged for protecting the confidentiality of the remote administration session	Volatile storage	OpenSSH package is used but all keys are overwritten with zeros before freeing memory
	SSH PKA	Public key authentication for remote administration over SSH	Non-volatile storage/file system	Overwrite with zeros to clear cache and read verify, then erase file.
	x509 certificate with keys	For TLS connections	Non-volatile storage/file system	Overwrite with zeros to clear cache and read verify, then erase file
	Local user password	User login	Non-volatile storage/file system (shadow file)	Erase file. Password is hashed with sha512, and the password file is only readable by root
	TLS session HMAC keys	For TLS connections with Audit server	Volatile Storage	OpenSSL package is used as infrastructure, but all keys are overwritten with zeros before freeing memory
	The evaluator examined the section titled **Cryptographic Key and CSP Destruction** in the Security Target to verify that the TSS description of keys and storage locations is consistent with the functions carried out by the TOE. Upon investigation, the evaluator found that the TSS states that: **The TOE meets all requirements specified in FIPS 140-2 for destruction of keys and Critical Security Parameters (CSPs).** Based on these findings, this assurance activity is considered satisfied.			
Verdict	Pass.			

5.2.3.2 FCS_CKM.4 TSS 2

Objective	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 examined the section 6 titled **TOE Summary Specifications** and the section titled **Cryptographic Key and CSP Destruction** in the Security Target to verify that 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. Upon investigation, the evaluator found that the TSS states that plaintext keys stored in the non-volatile storage are maintained by the file system. These plaintext keys are destroyed by SAOS when an administrator makes configuration changes (ssh server config or tls-service-profile config) that result in changing, replacing or deletion of these files. The keys are destroyed by the SAOS by overwriting the storage location of the key with a single overwrite of zeroes. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.2.3.3 FCS_CKM.4 TSS 3

Objective	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 evaluator examined the section 6.4 titled **Cryptographic Key and CSP Destruction** in the Security Target to verify 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. Upon investigation, the evaluator found that the TOE stores all keys in plaintext form. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.2.3.4 FCS_CKM.4 TSS 4

Objective	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 examined section 6.1 titled "**Fulfillment of the Security Functional Requirements**" in the Security Target to verify that the TSS identifies any configurations or circumstances that may not conform to the key destruction requirement. Upon investigation, the evaluator found that the only situation where the key destruction may be prevented would be if the system suffers a crash or loss of power. This situation only impacts the keys that are stored in

	the filesystem. Since the TOE is inaccessible in this situation, administrative zeroization cannot be performed. The keys stored in filesystem are not directly accessible to any user or administrator. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.2.3.5 FCS_CKM.4 TSS 5

Objective	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 TOE destroys plaintext cryptographic keys stored in the volatile storage by a single overwrite with zeroes. Plaintext keys stored in the non-volatile storage are destroyed by the SAOS overwriting the storage location of the key with a single overwrite of zeroes. The destruction of each cryptographic key and Critical Security Parameter (CSP) is summarized in table 15 of the ST. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.2.4 FCS_COP.1/DataEncryption

5.2.4.1 FCS_COP.1/DataEncryption TSS 1

Objective	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 section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS to ensure it identifies the key size(s) and mode(s) supported by the TOE for data encryption/decryption. Upon investigation, the evaluator found that the TSS states that: **The TOE implements symmetric encryption and decryption using AES in CBC, GCM and CTR modes. Key sizes of 128 and 256 bits are implemented. AES encryption and decryption is used by TLS and SSH protocols.** Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass.

5.2.5 FCS_COP.1/SigGen

5.2.5.1 FCS_COP.1/SigGen TSS 1

Objective	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 section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS to ensure it specifies the cryptographic algorithm and key size supported by the TOE for signature services. Upon investigation, the evaluator found that the TSS states that: **The TOE generates and verifies digital signatures with RSA Digital Signature Algorithm and cryptographic key sizes (modulus) 2048 bits and 4096 bits.** **The TOE also generates and verifies digital signatures with ECC using key sizes 256 bits for NIST curves P-256.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.2.6 FCS_COP.1/Hash

5.2.6.1 FCS_COP.1/Hash TSS 1

Objective	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 examined the section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS documents the association of the hash function with other TSF cryptographic functions. Upon investigation, the evaluator found that the TSS states that: **The TOE implements cryptographic message digest (hash value) computation using SHA-1, SHA-256, SHA-384, and SHA-512 with message digest sizes of 160 bits, 256 bits, 384 bits, and 512 bits respectively. The hashing algorithms are used in SSH and TLS connections for secure communications.** **The TOE uses message digests for the following functions:**

Function	SHA-1	SHA-256	SHA-384	SHA-512

		Digital signature computation		X	X	X
		Digital Signature verification	X	X	X	X
		TLS HMAC	X	X	X	
		SSH HMAC	X	X		X
		Password storage				X
		NTP Message Authentication	X			
	Based on these findings, this assurance activity is considered satisfied.					
Verdict	Pass.					

5.2.7 FCS_COP.1/KeyedHash

5.2.7.1 FCS_COP.1/KeyedHash TSS 1

Objective	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 section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS specifies the following values used by the HMAC function: key length, hash function used, block size, and output MAC length used. Upon investigation, the evaluator found that the TSS states that: **The HMAC (keyed Hash) algorithms used by the TOE are summarized in the following. For each HMAC, the table states the Hash algorithm used, the key size and the message digests (output) length.**

	Hash Algorithm	Key Size	Block Size/Output Length
HMAC-SHA-1	SHA-1	128 bits	160 bits
HMAC-SHA-256	SHA-256	256 bits	256 bits
HMAC-SHA-384	SHA-384	384 bits	384 bits
HMAC-SHA-512	SHA-512	512 bits	512 bits

	Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.2.8 FCS_RBG_EXT.1/ARMA53

5.2.8.1 FCS_RBG_EXT.1 TSS 1

Objective	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 section 6 titled **TOE Summary Specification** in the Security Target to verify that the TSS 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. Upon investigation, the evaluator found that the TSS states that: The TOE implement below random bit generator: CTR_DRBG (AES-256) implemented by OpenSSL and seeded by 256 bits of data read from the Kernel DRBG. The entropy sources used by the TOE depend on the exact variation of the TOE: • TOE variations 3926, 3928 and 3948 are implemented using ARM Cortex A53 with no readily available hardware sources of entropy. They harvest entropy from the Linux kernel v5.4 functions which accumulate and make available to other processes entropy from CPU jitter and CPU interrupts. • TOE variations 5144 and 5164 use the ARM Cortex A72 which implements the IP-76 hardware source of entropy. They also harvest entropy from the Linux kernel v5.4 functions which accumulate and make available to other processes entropy from CPU jitter and CPU interrupts. • TOE variations 5162, 5170, 5171, 8180, and the Large NFV Compute Servers (FRU) are implemented on Intel platforms which implement the RDRAND command that is used for reading entropy from a dedicated CPU circuitry, and also include Infineon SLM9670 or Infineon SLB9665 TPMv2.0 chip which implements a source of entropy which can be read by external processes. They also harvest entropy from the Linux kernel v5.4 functions which accumulate and make available to other processes entropy from CPU jitter and CPU interrupts.

	Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.2.9 FCS_RBG_EXT.1/ARMA72

5.2.9.1 FCS_RBG_EXT.1 TSS 1

Objective	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 section 6 titled **TOE Summary Specification** in the Security Target to verify that the TSS 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. Upon investigation, the evaluator found that the TSS states that: The TOE implements one random bit generator: CTR_DRBG (AES-256) implemented by OpenSSL and seeded by 256 bits of entropy. The entropy sources used by the TOE depend on the exact variation of the TOE: • TOE variations 3926, 3928 and 3948 are implemented using ARM Cortex A53 with no readily available hardware sources of entropy. They harvest entropy from the Linux kernel v5.4.154 functions which accumulate and make available to other processes entropy from CPU jitter and CPU interrupts. • TOE variations 5144 and 5164 use the ARM Cortex A72 which implements the IP-76 hardware source of entropy. They also harvest entropy from the Linux kernel v5.4.154 functions which accumulate and make available to other processes entropy from CPU jitter and CPU interrupts. • TOE variations 5162, 5170, 5171, 8180, and the Large NFV Compute Servers are implemented on Intel platforms which implement the RDRAND command that is used for reading entropy from a dedicated CPU circuitry, and also include Infineon SLM9670 or Infineon SLB9665 TPMv2.0 chip which implements a source of entropy which can be read by external processes. They also harvest entropy from the Linux kernel v5.4.154 functions which accumulate and make available to other processes entropy from CPU jitter and CPU interrupts. Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass.

5.2.10 FCS_RBG_EXT.1/Intel

5.2.10.1 FCS_RBG_EXT.1 TSS 1

Objective	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 section 6 titled **TOE Summary Specification** in the Security Target to verify that the TSS 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. Upon investigation, the evaluator found that the TSS states that: The TOE implement below random bit generator: CTR_DRBG (AES-256) implemented by OpenSSL and seeded by 256 bits of data read from the Kernel DRBG. The entropy sources used by the TOE depend on the exact variation of the TOE: • TOE variations 3926, 3928 and 3948 are implemented using ARM Cortex A53 with no readily available hardware sources of entropy. They harvest entropy from the Linux kernel v5.4 functions which accumulate and make available to other processes entropy from CPU jitter and CPU interrupts. • TOE variations 5144 and 5164 use the ARM Cortex A72 which implements the IP-76 hardware source of entropy. They also harvest entropy from the Linux kernel v5.4.154 functions which accumulate and make available to other processes entropy from CPU jitter and CPU interrupts. • TOE variations 5162, 5170, 5171, 8180, and the Large NFV Compute Servers are implemented on Intel platforms which implement the RDRAND command that is used for reading entropy from a dedicated CPU circuitry, and also include Infineon SLM9670 or Infineon SLB9665 TPMv2.0 chip which implements a source of entropy which can be read by external processes. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.3 TSS Activities (HTTPS)

5.3.1 FCS_HTTPS_EXT.1

5.3.1.1 FCS_HTTPS_EXT.1.1 TSS 1

Objective	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 section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS provides enough detail to explain how the implementation complies with RFC 2818. Upon investigation, the evaluator found that the TSS states that: **The TOE supports secure communication of the TOE with the File server over an HTTPS connection using TLS v1.2 implementation. In this scenario, the TOE acts as a TLS client communicating with the servers in the Operational Environment. The HTTPS protocol complies with RFC 2818.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.4 TSS Activities (NTP)

5.4.1 FCS_NTP_EXT.1

5.4.1.1 FCS_NTP_EXT.1 TSS 1

Objective	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 section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS 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. Upon investigation, the evaluator found that the TSS states that:

	The TOE supports the use of the NTP version 4 (NTP v4) to synchronize time with an NTP server. The TOE validates the integrity of the time source using SHA1 as the message digest algorithm. The TOE supports at least three and a maximum of ten NTP servers, however in the evaluated configuration up to 3 are claimed. **The TSF shall not update NTP timestamp from broadcast and/or multicast addresses. Both parameters are configured by default to not allow an update.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.4.1.2 FCS_NTP_EXT.1 TSS 2

Objective	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 evaluator examined the section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS describes each method selected in the ST, 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. Upon investigation, the evaluator found that the TSS states that: **The TOE supports the use of the NTP version 4 (NTP v4) to synchronize the clock of the TOE with an NTP server. The TOE validates the integrity of the time source using SHA1 as the message digest algorithm. The TOE supports at least three and a maximum of ten NTP servers, however in the evaluated configuration up to 3 are claimed.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.5 TSS Activities (SSH)

5.5.1 FCS_SSHS_EXT.1

5.5.1.1 FCS_SSHS_EXT.1.2 TSS 1 [TD0631]

Objective	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). 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. 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	The evaluator examined the section 6 titled **TOE Summary Specifications** in the Security Target to verify 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). Upon investigation, the evaluator found that the TSS states that: **The TOE implements both public key authentication and password-based authentication. Public key authentication methods supported are ssh-rsa and ecdsa-sha2-nistp256. Any other authentication algorithm requests are rejected. For Public key authentication, the TOE stores a user's (or SSH client's) public key in its local filesystem and associates that key with the user's identity. When a user or client presents its public key, the TOE matches it against the stored value to verify the user's identity. The password-based authentication acts as a fallback option in case the public key authentication fails.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.5.1.2 FCS_SSHS_EXT.1.3 TSS 1

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

Evaluator Findings	The evaluator examined section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS describes how "large packets" in terms of RFC 4253 are detected and handled. Upon investigation, the evaluator found that the TSS states that: **The TOE examines all packets for size and drops any packets greater than 32768 bytes and drops in accordance with RFC 4253.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.5.1.3 FCS_SSHS_EXT.1.4 TSS 1

Objective	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. 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 examined section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS specifies the optional characteristics and the encryption algorithms supported. Upon investigation, the evaluator found that the TSS states that: **For symmetric encryption, the TOE allows aes128-ctr, aes256-ctr, aes128-gcm@openssh.com and aes256-gcm@openssh.com. Requests for any other algorithms are rejected.** The evaluator found that the list corresponds to the selections in the corresponding SFR in the ST. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.5.1.4 FCS_SSHS_EXT.1.5 TSS 1 [TD0631]

Objective	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 examined the section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS specifies the optional characteristics and the public key algorithms supported. Upon investigation, the evaluator found that the TSS states that:

	The TOE implements a SSH Server for remote administrators to connect securely to the TOE and use the CLI from a remote management station. The TOE implementation of SSHv2 is in compliance with RFCs 4251, 4252, 4253, 4254, 4256, 4344, 5656, 6668, 8308 section 3.1, 8332. **The TOE implements both public key authentication and password-based authentication. Public key authentication methods supported are ssh-rsa and ecdsa-sha2-nistp256. Any other authentication algorithm requests are rejected.** The evaluator found that the list corresponds to the selections in the corresponding SFR in the ST. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.5.1.5 FCS_SSHS_EXT.1.6 TSS 1

Objective	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 examined section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS lists the supported data integrity algorithms, and that that list corresponds to the list in this component. Upon investigation, the evaluator found that the TSS states that: **For message authentication, the TOE allows hmac-sha1, hmac-sha2-256, hmac-sha2-512 and implicit. Requests for any other algorithms is rejected. Message authentication algorithm implicit is used for the @openssh.com symmetric encryption algorithms.** The evaluator found that the list corresponds to the selections in the corresponding SFR in the ST. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.5.1.6 FCS_SSHS_EXT.1.7 TSS 1

Objective	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 examined the section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS lists the supported key exchange algorithms, and that that list corresponds to the list in this component. Upon investigation, the evaluator found that the TSS states that:

	The SSHv2 implementation of the TOE enforces to only allow the diffie-hellman-group14-sha1, ecdh-sha2-nistp256, diffie-hellman-group14-sha256, diffie-hellman-group16-sha512, ecdh-sha2-nistp384, ecdh-sha2-nistp521 key exchange methods. The evaluator found that the list corresponds to the selections in the corresponding SFR in the ST. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.5.1.7 FCS_SSHS_EXT.1.8 TSS 1

Objective	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 examined section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS specifies that both thresholds are checked, and that rekeying is performed upon reaching the threshold that is hit first. Upon investigation, the evaluator found that the TSS states that: **The TOE is capable of rekeying. The TOE verifies the following thresholds:** • **No longer than one hour** • **No more than 1 gigabyte of transmitted data** **The TOE continuously checks both conditions. When either of the conditions are met, the TOE will initiate a rekey. The TOE can also be configured to ensure that SSH re-key of no longer than one hour and no more than one gigabyte of transmitted data for the session key.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.6 TSS Activities (TLS)

5.6.1 FCS_TLSC_EXT.1

5.6.1.1 FCS_TLSC_EXT.1.1 TSS 1

Objective	The evaluator shall check the description of the implementation of this protocol in the TSS to ensure that the ciphersuites supported are specified. The evaluator shall check the TSS to ensure that the ciphersuites specified include those listed for this component.
Evaluator Findings	The evaluator examined the section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS specifies the ciphersuites supported and that the ciphersuites specified include those listed for this component. Upon investigation, the evaluator found that the TSS states that: **The TOE implements a TLS Client which supports TLS 1.2 (RFC 5246) and rejects all other TLS and SSL versions. The TLS implementation supports the following ciphersuites:** • **TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268** • **TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268** • **TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492** • **TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492** • **TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246** • **TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246** • **TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288** • **TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288** • **TLS_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 checked the TSS and found that the ciphersuites specified include those listed for this component.

	Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.6.1.2 FCS_TLSC_EXT.1.2 TSS 1

Objective	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 examined the section 6 titled **TOE Summary Specifications** in the Security Target to verify 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; whether IP addresses and wildcards are supported. Upon investigation, the evaluator found that the TSS states that: **The TOE uses TLS for all trusted channels and also to implement HTTPS. Peer entities are authenticated with the X.509 certificates. The trusted channel is established when the peer certificate is valid. The TOE verifies that the presented identifier matches the reference identifier in order to establish the connection.** **The TOE supports SAN extension and checks SAN extension over CN when present. The TOE ignores CN when SAN is present. When SAN is not present, the TOE falls back to CN check. FQDN is supported in both SAN and CN while IP address is only supported in SAN.** The TOE supports wildcards in certificates. The wildcard must be in the left-most label of the presented identifier and can only cover one level of subdomains. For the reference identifier without a left-most label as in the certificate, the connection will fail, i.e., awesome.com doesn't match *.awesome.com. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.6.1.3 FCS_TLSC_EXT.1.2 TSS 3

Objective	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. 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 examined the section 6 titled **TOE Summary Specifications** in the Security Target to verify that, if IP addresses are supported in the CN as reference identifiers, 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 and whether canonical format is enforced. Upon investigation, the evaluator found that the TSS states that: **FQDN is supported in both SAN and CN while IP address is only supported in SAN.** Since IP addresses are not supported in the CN as a reference identifier, this assurance activity is satisfied implicitly. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.6.1.4 FCS_TLSC_EXT.1.4 TSS 1

Objective	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 examined section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS describes the Supported Elliptic Curves Extension and whether the required behaviour is performed by default or may be configured. Upon investigation, the evaluator found that the TSS states that: **The 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. The TOE will validate the server's certificate according to FIA_X509_EXT.1/Rev. If the server certificate is invalid, the connection will not be established.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.7 TSS Activities (Identification and Authentication)

5.7.1 FIA_AFL.1

5.7.1.1 FIA_AFL.1 TSS 1

Objective	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 examined section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS contains a description, for each supported method for remote administrative actions, of how successive unsuccessful authentication attempts are detected and tracked; the method by which the remote administrator is prevented from successfully logging on to the TOE; and the actions necessary to restore this ability. Upon investigation, the evaluator found that the TSS states that: **The TOE maintains a counter of consecutive failed authentication attempts for each user. The counter tracks the number of failed authentication attempts for all remote authentication attempts.** **When an authentication fails, the counter value is incremented. When an authentication succeeds, the counter value is reset. The maximum number of allowed consecutive authentication attempts may be set by the administrator of the TOE. When the maximum number is reached, the TOE shall lock the account and start a session lockout timer for the account. Once the lockout timer expires, the TOE shall unlock the account. The duration of the lockout may be set by the administrator of the TOE**. Based on these findings, this assurance activity is considered satisfied
Verdict	Pass.

5.7.1.2 FIA_AFL.1 TSS 2

Objective	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).

Evaluator Findings	The evaluator examined section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS ensures that authentication failures by remote administrators cannot lead to a situation where no administrator access is available. Upon investigation, the evaluator found that the TSS states that: **Accounting locking only applies to remote authentication attempts. Even if an account is locked, the same account may still be used from console if the user is successfully authenticated. This ensures that at no time shall the TOE be in a state where each administrator is locked out and no administrator access to the TOE is possible.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.7.2 FIA_PMG_EXT.1

5.7.2.1 FIA_PMG_EXT.1.1 TSS 1 [TD0792]

Objective	The evaluator shall check that the TSS lists the supported special character(s) for the composition of administrator passwords. The evaluator shall check the TSS to ensure that the minimum_password_length parameter is configurable by a Security Administrator. 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 evaluator examined section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS contains the lists of the supported special character(s) and minimum and maximum number of charters supported for administrator passwords. Upon investigation, the evaluator found that the TSS states that: • **Passwords can be composed of any combination of upper and lower case letters, numbers, and the following special characters: "!", "@", "#", "$", "%", "^", "&", "*", "(","")", "", "+", ",", "-", ".", "/", ":", ";", "<", "=", ">", "[", "\", "]", "_", "`", "{", "}"and "~".;** • **The minimum length of a password may be configured by the administrator and can be any integer value between 1 and 128 (inclusive).** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.7.3 FIA_UIA_EXT.1

5.7.3.1 FIA_UIA_EXT.1 TSS 1

Objective	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 section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS describes the logon process for each logon method supported for the product. Upon investigation, the evaluator found that the TSS states that: **Administrative access to the TOE is facilitated through one of several interfaces:** • **Directly connecting to the Ciena appliance locally from a console connected to the serial port of a USB-C port of the TOE.** • **Remotely connecting to the Ciena appliance via SSHv2.** **For local access, the TOE prompts the user to enter a username and password, then compares the entered password to the reference password stored for the user. If the verification succeeds and the user is allowed to enter the role administrator, the TOE assigns the user to the role administrator and grants access to the CLI. If the username does not exist or the password is incorrect, the TOE denies the access and returns to the authentication window to request a username and password.** **For remote access, the TOE may be configured to require RSA public key authentication or password-based authentication. The remote access is implemented using SSH. Successful authentication occurs when either the cryptographic authentication protocol is successfully completed between the TOE and the remote management workstation, or the password verification succeeds in a manner identical to the authentication for local access.** **If the authentication is successful and the user is granted access to the role administrator, the TOE establishes a SSH connection between the remote management workstation and the TOE and grants the user administrator rights to the TOE (i.e., makes available the CLI). If the authentication fails, the TOE denies access and returns to the authentication windows.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.7.3.2 FIA_UIA_EXT.1 TSS 2

Objective	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 section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS describes which actions are allowed before user identification and authentication. Upon investigation, the evaluator found that the TSS states that: **The TOE requires all users to be successfully identified and authenticated prior to assigning them to the role administrator and granting them access to the TOE. The TOE displays an access banner to each user prior at the identification and authentication window.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.7.6 FIA_X509_EXT.1/Rev

5.7.6.1 FIA_X509_EXT.1/Rev TSS 1

Objective	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 examined section 6 titled **TOE Summary Specifications** in the Security Target to verify that 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). Also, upon investigation, the evaluator found that the TSS states that the certificate validity and revocation checking is performed to support authentication of external TLS peers such as audit server and file server. The certificate validity and revocation checking is performed on each certificate when it is uploaded into the TOE as well as part of the authentication step. The TOE used OCSP for revocation checking.

	Also, upon investigation, the evaluator found that the TSS states that: **The TOE also 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 will only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.** **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 validity and revocation checking is performed to support authentication of external TLS peers such as audit server and file server. The certificate validity and revocation checking is performed on each certificate when it is uploaded into the TOE as well as part of the authentication step. The TOE used OCSP for revocation checking. If the validation of the certificate fails because the OCSP Server cannot be connected to, the certificate shall not be accepted, and the connection is terminated.** Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass.

5.7.6.2 FIA_X509_EXT.1/Rev TSS 2

Objective	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 evaluator examined section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS describes when revocation checking is performed and on what certificates. Upon investigation, the evaluator found that the TSS states that the **certificate validity and revocation checking is performed to support authentication of external TLS peers such as audit server and file server. The certificate validity and revocation checking are performed on each certificate when it is uploaded into the TOE as well as part of the authentication step. The TOE used OCSP for revocation checking.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.7.7 FIA_X509_EXT.2

5.7.7.1 FIA_X509_EXT.2 TSS 1

Objective	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 examined section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS describes how the TOE chooses which certificates to use. 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 HTTPS and TLS peer entities. Certificates are used to authenticate and establish secure communication channel for the File Server and syslog servers. The TOE supports RSA based certificates in PKCS#12.**

	The TOE allows each TLS service to be configured with its own certificate in a TLS profile. Once a certificate is configured for a Syslog Server using a TLS profile, that certificate will be used for all Syslog Server connection authentication. Likewise, once a certificate is configured for TLS File Server, that certificate will be used for all TLS File Server connection authentication. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.7.7.2 FIA_X509_EXT.2 TSS 2

Objective	The evaluator shall examine the TSS to confirm 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 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 guidance documentation contains instructions on how this configuration action is performed.
Evaluator Findings	The evaluator examined the section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS 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. Upon investigation, the evaluator found that the TSS states that: **The TOE will check the validity of the TLS Server certificate prior to establishing a TLS connection with the TLS server. The certificate validation is determined based on reference ID verification, certificate path, extendedKeyUsage field, certificate expiry date and the certificate revocation status. Both trusted channels are treated the same.** **The TOE used OCSP for revocation checking. If the validation of the certificate fails because the OCSP Server cannot be connected to, the certificate shall not be accepted, and the connection is terminated.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.8 TSS Activities (Security Management)

5.8.1 FMT_MOF.1/Functions Management of security functions behaviour

5.8.1.1 FMT_MOF.1/Functions TSS 2

Objective	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).
Evaluator Findings	The evaluator examined the section 6 titled "**TOE Summary Specifications"** in the Security Target to verify that the TSS identifies 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). Upon investigation, the evaluator found that the TSS states that: **The TOE restricts the ability to enable and disable the transmission of audit records to an external audit server to Security Administrators.** **The Security Administrator can modify the behavior of transmitting audit data to an external audit server with the following configuration from the CLI:** ***config*** **syslog log-actions remote-syslog-tls destination <ip address or DNS>** **syslog log-actions remote-syslog-tls admin-state <enable/disable>** **The TOE's login users can view what is configured from the CLI:** **show syslog tls** **show syslog tls statistics** Also, the same section states that when the local audit data store capacity is exhausted, the TOE will overwrite audit records starting with the oldest audit record. Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass.

5.8.2 FMT_MTD.1/CoreData Management of TSF Data

5.8.2.1 FMT_MTD.1/CoreData TSS 1

Objective	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 examined section 6 titled "**TOE Summary Specifications"** in the Security Target to verify that the TSS identifies administrative functions that are accessible through an interface prior to administrator log-in. Upon investigation, the evaluator found that the TSS states that: **The only access the TOE allowed prior to the successful identification and authentication of the user is the access banner displayed at each login prompt.** The evaluator examined the section titled "**TOE Summary Specifications"** in the Security Target to verify that the TSS details how the ability to manipulate the TSF data through these interfaces is disallowed for non-administrative users. Upon investigation, the evaluator found that the TSS states that: **The TOE restricts the ability to manage the TOE to Security Administrators. This is achieved via role-based access control and privileges assigned to each role.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.8.2.2 FMT_MTD.1/CoreData TSS 2

Objective	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 evaluator examined the section 6 titled "**TOE SUMMARY SPECIFICATION"** in the Security Target to verify that, if the TOE supports handling of X.509v3 certificates and implements a trust store, the TSS contains sufficient information to

	describe how the ability to manage the TOE's trust store is restricted. Upon investigation, the evaluator found that the TSS states that: **The TOE restricts the ability to manage SSH (public keys), TLS (public keys), and any configured X.509 certificates (public key) to the security administrators. This is achieved via role-based access control and privileges assigned to each role.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.8.3 FMT_MTD.1/CryptoKeys Management of TSF Data

5.8.3.1 FMT_MTD.1/CryptoKeys TSS 2

Objective	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 section 6 titled "**TOE Summary Specifications"** in the Security Target to verify that 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. Upon investigation, the evaluator found that the TSS states that: **Only the Security Administrator has the ability to configure the authentication keys TLS functionality and can modify, import, and delete the key for SSH.** **The TOE restricts the ability to manage SSH (public keys), TLS (public keys), and any configured X.509 certificates (public key) to the security administrators. This is achieved via role-based access control and privileges assigned to each role.** **The Security Administrator manages the SSH public key by configuring the public-key-based authentication keys for users:**

	system ssh-server user-pubkey install user-name <user> filename <filename> [server-type <server-type>] address <address> [login <login-id> password <password>] **OR** **system ssh-server user-pubkey install user-name <user> url <url>** **The Security Administrator manages the X.509 certificates used in TLS communication with the peers by configuring the TLS profile, associating the peer authentication profile and importing TLS certificate to the profile:** **1. Identify the TLS Profile** **tls-service-profiles <tls-service-profile-name> tls-profile-name <tls-profile>** **2. Identify the peer authentication profile.** **tls-service-profiles <tls-service-profile-name> tls-peer-auth-profile-name <peer-auth-profile>** **3. Identify the TLS certificate.** **tls-service-profiles <tls-service-profile-name> tls-certificate-name <certificate>** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.8.4 FMT_SMF.1 Specification of Management Functions

5.8.4.1 FMT_SMF.1 TSS 1

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

	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, section 6.1 in ST, along with the guidance documentation, and observed the TOE during testing activities. It was confirmed that the management functions specified in FMT_SMF.1 in ST section 5.3.4.5 were provided by the TOE. The evaluator examined section 6 titled "**TOE Summary Specifications"** in the Security Target to verify that it details which security management functions are available through which interface(s). Upon investigation, the evaluator found that the TSS states that: **The TOE implements a management interface for the Security Administrators to configure the TOE. The management interface is a Command Line Interface (CLI) which may be accessed locally from a management workstation connected to the TOE on the console or USB-C interface, or from a remote management workstation connected to the TOE network management port over SSH.** The evaluator examined the section titled "**the TOE and the Operational Environment**" in the AGD to verify that they describe the local administrative interface. Section 2.3 of the AGD also states that "The management interface is a Command Line Interface (CLI) which may be accessed locally or remotely. Local access is via a console port which is a Serial EIA-561 (RJ-45) or a USB-C port. It allows management of the TOE from a workstation physically connected to the TOE. Remote management is over Secure Shell (SSH)". Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.8.5 FMT_SMR.2 Restrictions on Security Roles

5.8.5.1 FMT_SMR.2 TSS 1

Objective	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 section 6 titled "**TOE SUMMARY SPECIFICATION"** in the Security Target to verify 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. Upon investigation, the evaluator found that the TSS states that: **The TOE only implements a single administrative role: Security Administrator. Users that belong to "Super" or "admin" groups have administrative privileges and assume the role of Security Administrator. Users may, upon successful authentication, based on their privileges (i.e. "Super" or "admin" groups), enter the role of Security Administrators locally and remotely and be granted access to the CLI which allows Security Administrators to manage the TOE locally and remotely. The TOE also supports a single non-administrative role: Read-Only User. Users that belong to "Limited" group have read-only privileges. Read-Only User cannot make any changes to the TOE configuration.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.9 TSS Activities (Protection of the TSF)

5.9.1 FPT_APW_EXT.1 Protection of Administrator Passwords

5.9.1.1 FPT_APW_EXT.1 TSS 1

Objective	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 section 6 titled "**TOE Summary Specifications"** in the Security Target to verify that the TSS details all authentication data that are subject to this requirement and the method used to obscure the plaintext password data when stored. Upon investigation, the evaluator found that the TSS states that: **All passwords are stored by the TOE hashed and salted using SHA-512. The storage and management of the passwords is implemented using the standard Linux Pluggable Authentication Mechanism (PAM) functions. The passwords are stored in a way such that they cannot be viewed by any user.**

	Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

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

5.9.2.1 FPT_SKP_EXT.1 TSS 1

Objective	The evaluator shall examine the TSS to determine that it details how any preshared 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 section 6 titled "**TOE Summary Specifications"** in the Security Target to verify that the TSS 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. Upon investigation, the evaluator found that the TSS states that: **Pre-shared keys, symmetric keys, and private keys are stored encrypted. An exception is the SSH host keys which are required by the SSH Daemon when the SSH starts. The SSH Daemon is executed at the root privileges and the SSH host keys are only accessible with root privileges. During the setup and configuration of the TOE when cryptographic keys are generated, the TOE stores all private keys in a secure directory that is not readily accessible to administrators.** **The TOE can only be accessed through the CLI which implements the complete management interface of the TOE. The CLI does not implement any functions for displaying the symmetric keys, asymmetric private keys, passwords, or any other secret parameters.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.9.3 FPT_STM_EXT.1 Reliable Time Stamps

5.9.3.1 FPT_STM_EXT.1 TSS 1[TD0632]

Objective	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. 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 evaluator examined the section 6 titled "**TOE Summary Specification"** in the Security Target to verify that the TSS lists each security function that makes use of time and provides a description of how the time is maintained and considered reliable in the context of each of the time related functions. Upon investigation, the evaluator found that the TSS states that: **The TOE implements a hardware clock for local date and time. The clock may also be configured to update the time from an external NTP server. The time is maintained by the TOE either via internal hardware clock or connecting to an external NTP server. The hardware clock is a real-time clock (RTC) with battery to maintain time across reboots and power loss. If an NTP server is configured, the TOE synchronizes the time with the external NTP server periodically. The time is used for producing time stamps which are attached to audit records and to check the X.509 certificate expiration. The TOE also uses the clock to implement the session time out timers for each interactive session and to terminate each interactive session which exceeds the maximum allowed inactivity time.** The selection "obtain time from the underlying virtualization system" is not selected in ST. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

5.9.4 FPT_TST_EXT.1.1 TSF testing

5.9.4.1 FPT_TST_EXT.1.1 TSS 1

Objective	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). 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 examined section 6 titled "**TOE summary specification"** in the Security Target to verify that the TSS details the self-tests that are run by the TSF on start-up. Upon investigation, the evaluator found that the TSS states that: **The TSF runs the following self-tests during initial start-up (on power on)** • **Check of various FPGA devices access and sanity,** • **Check of PCI bus and devices response,** • **FileSystem Integrity** • **Crypto KAT/self-test**. **The "*Check of various FPGA devices access and sanity*" self-test verifies the health, integrity and accessibility of all hardware FPGAs within the TOE by comparing the value in FPGA's scratchpad.** **The "Check of PCI bus and devices response" self-test verifies the status of PCI bus and its connectivity to various PCIe devices connected including its backplane by sending a signal to the connected PCI device and receiving a response back.** **The "Filesystem Integrity" self-test verifies the firmware integrity by computing a hash and verifying with its stored value.** **The "Crypto KAT/self-test" self-test verifies the FIPS know answer tests for various algorithms implemented by the TOE's cryptographic library such as AES, SHA, RSA, ECDSA, DRBG, and HMAC**. The evaluator examined section 6 titled "**TOE summary specification"** in the Security Target to verify that the TSS makes an argument that the tests are sufficient to demonstrate that the TSF is operating correctly. Upon investigation, the evaluator found that the TSS states that:

	The TOE executes self-tests during initial start-up to ensure correct operation and enforcement of its security functions. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.9.5 FPT_TUD_EXT.1 Trusted Update

5.9.5.1 FPT_TUD_EXT.1 TSS 1

Objective	The evaluator shall verify that the TSS describe 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 examined section 6 titled "**TOE Summary Specifications"** in the Security Target to verify that the TSS describes how to query the currently active version. Upon investigation, the evaluator found that the TSS states that: **The TOE provides authenticated users the ability to query the currently executing version of the TOE firmware with the show software command.** The evaluator examined the section titled "**TOE Summary Specifications"** in the Security Target to verify that the TSS, if a trusted update can be installed on the TOE with a delayed activation, describes how and when the inactive version becomes active. Upon investigation, the evaluator found that the TSS states that: **The TOE does not support partial upgrades, the TOE software shall at each upgrade be upgraded in its entirety. The TOE does not support delayed activation.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.9.5.2 FPT_TUD_EXT.1 TSS 2

Objective	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).

	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. 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 examined the section 6 titled "**TOE Summary Specifications"** in the Security Target to verify that the TSS describes all TSF software update mechanisms for updating the system firmware and software, includes a digital signature verification of the software before installation and that installation fails if the verification fails. Upon investigation, the evaluator found that the TSS states that: **The TOE allows the Security Administrators to manually upgrade the TOE software to the version available at the vendor's web site. Upgrades are digitally signed using RSA with SHA-256. The signature will be verified by the TOE during an upgrade. Upon successful verification of the signature, the image will be loaded onto the TOE.** Also, the same section states the following: " **If the images cannot be verified, the image will not be loaded onto the TOE. The TOE does not support partial upgrades, the TOE software shall at each upgrade be upgraded in its entirety**." The evaluator examined the section titled "**TOE Summary Specifications"** in the Security Target to 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. Upon investigation, the evaluator found that the TSS states that: **The Security Administrator can obtain the software upgrade from the Ciena website and place it on a trusted filer server. The TOE may then be instructed to connect to the file server using HTTPS over TLS and install the software image.** **The CLI command for downloading and installing the software upgrade is:**

	`software install 'url information' tls-service-profile <tls-service-profile Name>.` If the images cannot be verified, the image will not be loaded onto the TOE. The TOE does not support partial upgrades, the TOE software shall at each upgrade be upgraded in its entirety. The TOE does not support delayed activation. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.9.5.3 FPT_TUD_EXT.1 TSS 3

Objective	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 evaluator reviewed the Security Target and noted that the options 'support automatic checking for updates' or 'support automatic updates' were not selected. In light of these observations, this assurance activity is deemed to be fulfilled
Verdict	Pass.

5.9.5.4 FPT_TUD_EXT.1 TSS 4

Objective	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	Published Hash option is not claimed. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.10 TSS Activities (TOE Access)

5.10.1 FTA_SSL_EXT.1 TSF-Initiated Session Locking

5.10.1.1 FTA_SSL_EXT.1 TSS 1

Objective	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 section 6 titled **TOE Summary Specifications** in the Security Target to verify that the TSS identifies whether local administrative session locking, or termination is supported and the related inactivity time period settings. Upon investigation, the evaluator found that the TSS states that: **The TOE will terminate a local interactive session after a configurable time interval of session inactivity. If a local user session is inactive for a configured maximum period of inactivity, the TOE will terminate the session.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.10.2 FTA_SSL.3 TSF-Initiated Termination

5.10.2.1 FTA_SSL.3 TSS 1

Objective	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 section 6 titled "**TOE Summary Specifications"** in the Security Target to verify that the TSS identifies administrative remote session termination and the related inactivity time period. Upon investigation, the evaluator found that the TSS states that: **The TOE will terminate a remote interactive session after a configurable time interval of session inactivity.** **If a remote administrative session is inactive for a configured maximum period of inactivity, the session will be terminated. Fresh identification and authentication shall be required for the creation of a new session. The session inactivity timer will be restored for the new session.**

	The Security Administrator may configure the TOE to terminate an inactive session after a specified period of time. The session timeout and inactive timeout values are indicated in seconds. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.10.3 FTA_SSL.4 User-Initiated Termination

5.10.3.1 FTA_SSL.4 TSS 1

Objective	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 section 6 titled "**TOE Summary Specifications"** in the Security Target to verify that the TSS identifies details how the local and remote administrative sessions are terminated. Upon investigation, the evaluator found that the TSS states that: **The TOE allows Administrators to terminate their own interactive sessions with the TOE using the exit command.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.10.4 FTA_TAB.1 Default TOE Access Banners

5.10.4.1 FTA_TAB.1 TSS 1

Objective	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). 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 examined the section 6 titled "**TOE Summary Specifications"** in the Security Target to verify that the TSS details each administrative method of access available to the Security Administrator and states that the TOE is displaying

	an advisory notice and consent warning message for each administrative method of access. Upon investigation, the evaluator found that the TSS states that: **The TOE implements an administrator-configurable access banner which is displayed at each login window. Both methods of accessing the TOE (locally form console and remotely over SSH) require user authentication. The access banner displaying an advisory notice and a consent warning message for each administrative method of access is displayed at each login prompt.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.11TSS Activities (Trusted Path/Channels)

5.11.1 FTP_ITC.1 Inter-TSF Trusted Channel

5.11.1.1 FTP_ITC.1 TSS 1

Objective	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. 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 examined section 6 titled "**TOE Summary Specifications"** in the Security Target to verify that the TSS, 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. Upon investigation, the evaluator found that the TSS states that: **The TOE implements a TLS Client for a trusted channel between itself and authorized IT entities. The remote entity may be an audit server or a File server. The TOE implements TLS between itself and a remote audit server and implements HTTPS over TLS for between itself and a remote file server.** The evaluator examined the section 6 titled "**TOE Summary Specifications"** in the Security Target to verify that the TSS describes all secure communication mechanisms in sufficient detail to allow the evaluator to match them to the

	cryptographic protocol Security Functional Requirements listed in the ST. Upon investigation, the evaluator found that the TSS states that: **The TOE implements a TLS Client for a trusted channel between itself and authorized IT entities. The remote entity may be an audit server or a File server. The TOE implements TLS between itself and a remote audit server and implements HTTPS over TLS for between itself and a remote file server.** **The file server is used in TOE software upgrades and storing user files. The audit server is used in exporting the audit logs securely.** **Each TLS and HTTPS/TLS connection is logically distinct from other communication channels and protects the channel data from disclosure and allows detection of the modification of the channel data. Peer entity authentication is performed using X.509 certificates for assured identification of the end points used in the trusted channels.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

5.11.2 FTP_TRP.1/Admin Trusted Path

5.11.2.1 FTP_TRP.1/Admin TSS 1

Objective	The evaluator shall examine the TSS to determine that the methods of remote TOE administration are indicated, along with how those communications are protected. 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 examined section 6 titled "**TOE Summary Specifications"** in the Security Target to verify that the TSS indicates the methods of remote TOE administration and how those communications are protected. Upon investigation, the evaluator found that the TSS states that: **The TOE implements a SSH server which allows SSHv2 connection between a remote management station and the TOE. A CLI which implements the management interface of the TOE is available to a remote administration over an encrypted SSHv2 channel. The remote administrator must initiate connection to the TOE using the SSH Client of the remote management station.**

	The evaluator examined the section 5.3.7.2 in the Security Target to verify that the protocol mentioned in the TSS section is consistent with those specified in the requirement. Upon investigation, the evaluator found that the TSS states that: **The TOE implements an SSH server which allows SSHv2 connection between a remote management station and the TOE.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6 Detailed Guidance Assurance Activities

6.1 Guidance Activities (Auditing)

6.1.1 FAU_GEN.1

6.1.1.1 FAU_GEN.1 Guidance 1

Objective	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 examined section titled **"Security Relevant Events"** in AGD[1] to verify that it provides an example of each auditable event required by FAU_GEN.1. Each event listed in the NDcPP is listed in AGD[1]. Next, the evaluator reexamined AGD[1] and found that the section titled "**Security Relevant Events**" contains a listing and description of each of the fields in generated audit records that contain the information required in FAU_GEN.1.2. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.1.1.2 FAU_GEN.1 Guidance 2

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

<table>
<tr><td></td><td>cPP. 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.</td></tr>
<tr><td>Evaluator Findings</td><td>The evaluator examined AGD[1] to verify that it identifies administrative commands, including subcommands, scripts, and configuration files, which 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 evaluator first examined the entirety of AGD[1] and the ST to determine what administrative commands are associated with each administrative activity. Upon investigation, the evaluator found that the following are applicable:
<table>
<tr><th>Administrative Activity</th><th>Method (Command Configuration)</th><th>AGD[1] Section Name</th></tr>
<tr><td>Startup of Audit function</td><td>syslog log-actions remote-syslog-tls admin-state enabled</td><td>Turn Logging On/Off</td></tr>
<tr><td>Shutdown of Audit function</td><td>syslog log-actions remote-syslog-tls admin-state disabled</td><td>Turn Logging On/Off</td></tr>
<tr><td>Logout</td><td>Exit</td><td>SSH Administrator Initiated Session Termination</td></tr>
<tr><td>Generating Keys (certificates)</td><td>pkix-certificates-csr-generate cert-name <cert_name> algorithm-identifier <algorithm-identifier> remote-fileuri ftp://server_ip/<path>/<cert.cnf> cert-passphrase <cert_passPhrase></td><td>Configure the Certificates Required for the TOE</td></tr>
<tr><td>Configuring Login failure threshold</td><td>system aaa authentication lockout-policy config fail-limit<integer></td><td>User Lockout</td></tr>
<tr><td>Display system information</td><td>show software</td><td>Verifying the TOE Version</td></tr>
</table></td></tr>
</table>

	Configuring CAs	pkix-ca install ca-cert-name <ca_cert_name> remote-fileuri scp://<server_ip>/<cert_path>/<ca.cert> login-id <login_id> password <login_password>	Configure the Certificates Required for the TOE
	Enable OCSP state	hello-params <profile-name> ocsp-state <state>	Modify the OCSP Default Responder URL
	Performing Software Updates	A series of CLI commands is provided for performing updates	Product Updates
	Setting the Time	system set clock <current datetime>	Clock Management
	Configuring Admin Timeout	system aaa authentication lockout-policy config lockouttime<integer>	SSH Idle Session Termination
	Configuring the Audit Server	syslog log-actions remote-syslog-tls destination "Syslog TLS server IP address or a DNS domain name."	Configuring Syslog
	Configuring Access Banner	system config motd-banner <banner-text>	"Login Banners"
	Setting Password Length	system aaa authentication password-policy config minlength <integer>	"Password Rules"
	Configuring SSH	system ssh-server config public-key-authentication enabled	Configuring the Remote Management Interface
	Next, the evaluator examined each of the test cases and identified test cases which exercised the above referenced functionality. The audit record associated with the configuration was captured. The following table identifies the test cases in which audit records for those configurations can be found.		

Administrative Activity	Test Case(s)
Startup	FAU_GEN.1
Shutdown	FAU_GEN.1
Login	FIA_UAU_EXT.2
Logout	FTA_SSL.4
Generating Keys (certificates)	FIA_X509_EXT.1.1/Rev Test #1a
Deleting Keys (certificates)	FMT_MTD.1/CryptoKeys
Resetting Passwords	FMT_SMF.1
Display system information	FAU_GEN.1 T1
Configuring CAs	FIA_X509_EXT.1.1/Rev T1
Performing Software Updates	FPT_TUD_EXT.1 T1
Setting the Time	FPT_STM_EXT.1 T1
Configuring Admin Timeout	FTA_SSL_EXT.1 T1
Configuring the Audit Server	FAU_GEN.1 T1
Configuring Access Banner	FTA_TAB.1 T1
Setting Password Length	FIA_PMG_EXT.1 T1
Configuring SSH	FCS_SSHS_EXT.1.4 T1

The above analysis illustrates that each of the relevant configuration methods is appropriately audited by the TOE.

Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass.

6.1.2 FAU_STG.1

6.1.2.1 FAU_STG.1 Guidance 1

Objective	The evaluator shall examine the guidance documentation to determine that it describes any configuration required for protection of the locally stored audit data against unauthorized modification or deletion.
Evaluator Findings	The evaluator examined the section titled "**Deleting Audit Records**" in AGD[1] to verify if any configuration is necessary to protect locally stored audit data against unauthorized modification or deletion. Upon investigation, the evaluator found that AGD[1] states that: **Only authorized administrators may view and clear audit records using the CLI which is the sole interface to the management functions of the TOE. Protected access to the local audit records is configured by default and therefore, does not need an administrator action at startup.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.1.3 FAU_STG_EXT.1

6.1.3.1 FAU_STG_EXT.1 Guidance 1

Objective	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 section titled "**Configuring Syslog**" and "**Configuring Log Level**" in AGD[1] to verify that 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. Upon investigation, the evaluator found that the AGD[1] states that: To configure syslog: Enable logging.

	`syslog log-actions remote-syslog-tls admin-state <enable/disable>`
	
	Identify the IP address of the syslog server.
	
	`Config syslog log-actions remote-syslog-tls destination <ip address or DNS>`
	Assign a TLS Profile to the connection.
	
	`syslog log-actions remote-syslog-tls tls-service-profile "profile name"`
	
	Set TLS timeout.
	
	`syslog log-actions remote-syslog-tls timeout 6`
	The parameter `tls-service-profile` points to a TLS Profile. Refer to section 6 of the AGD document, to create a TLS Profile.
	
	To configure log level:
	
	Steps
	
	1. Configure the severity of secure syslog messages.
	`syslog log-action remote-syslog-tls destination <address> severity <severity>`
	
	Example
	
	1. Configure the severity of secure syslog messages for the evaluated configuration.
	`syslog log-action remote-syslog-tls destination 10.1.5.200 severity alert critical debug emergency error info notice warning`
	Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass.

6.1.3.2 FAU_STG_EXT.1 Guidance 2

Objective	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 examined the section titled "**Local Logs**" in AGD[1] to verify that it describes the relationship between the local audit data and the audit data that are sent to the audit log server. Upon investigation, the evaluator found that the AGD states that: **The local log buffer is circular. By default, newer messages overwrite older messages after the buffer is full.** **When configured for a syslog backup the TOE will simultaneously offload events from a separate buffer to the external syslog server. This buffer is used to queue events to be sent to the syslog server if the connection to the server is lost. It is a circular buffer, so when the events overrun the storage space overwrites older events.** The evaluator determined that the TOE maintains two audit data buffers when syslog is configured. And that local audit data is saved locally, and audit data is sent to the syslog server simultaneously. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.1.3.3 **FAU_STG_EXT.1 Guidance 3**

Objective	The evaluator shall also ensure that the guidance documentation describes all possible configuration options for FAU_STG_EXT.1.3 and the resulting behavior of the TOE for each possible configuration. The description of possible configuration options and resulting behavior shall correspond to those described in the TSS.
Evaluator Findings	The evaluator examined the section titled "**Local Logs"** in AGD[1] to verify that it describes all possible configuration options for FAU_STG_EXT.1.3 and the resulting behavior of the TOE for each possible configuration. Upon investigation, the evaluator found that the AGD states that: **The local log buffer is circular. By default, newer messages overwrite older messages after the buffer is full.** The evaluator concluded that there is no configuration needed for the FAU_STG_EXT.1 SFR. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.2 Guidance Activities (Cryptographic Support)

Note that Test activities in the SD that are typically addressed by referencing CAVP certs are addressed in this section and are identified as "Test/CAVP" activities.

6.2.1 FCS_CKM.1

6.2.1.1 **FCS_CKM.1 Guidance 1**

Objective	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 examined the section titled "**Default Crypto Configuration**" in AGD[1] to verify that it 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. Upon investigation, the TOE is automatically installed in Default Crypto Configuration. **The TOE is automatically installed in Default Crypto Configuration. This means that a subset of the cryptographic library functions provided by the TOE are automatically configured to support the values identified in the Security Target.** Specifically: • **Supports the selected key generation scheme(s) and key size(s) for all cryptographic protocols defined in the Security Target (FCS_CKM.1).** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.2.3 FCS_CKM.2

6.2.3.1 FCS_CKM.2 Guidance 1

Objective	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 examined the section titled "**Default Crypto Configuration**" in AGD[1] to verify that it instructs the administrator how to configure the TOE to use the selected key establishment scheme(s). Upon investigation, the evaluator found that AGD[1] states that**:**

	The TOE is automatically installed in Default Crypto Configuration. This means that a subset of the cryptographic library functions provided by the TOE are automatically configured to support the values identified in the Security Target. Specifically: • **Supports the selected key establishment scheme(s) for all cryptographic protocols defined in the Security Target (FCS_CKM.2).** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.2.4 FCS_CKM.4

6.2.4.1 FCS_CKM.4 Guidance 1

Objective	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). The evaluator shall check that the guidance documentation provides guidance on situations where key destruction may be delayed at the physical layer.
Evaluator Findings	The evaluator examined the section titled "**Default Crypto Configuration**" in AGD[1] to verify that it 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. Upon investigation, the evaluator found that the AGD states that: **TOE destroys plaintext cryptographic keys stored in volatile storage by a single overwrite with zeroes. Plaintext keys stored in the non-volatile storage are destroyed by the SAOS overwriting the storage location of the key with a single overwrite of zeroes.** **The above key destruction methods apply to all configurations and circumstances, except one. The only situation where the key destruction may be prevented would be if the system suffers a crash or loss of power. This situation only impacts the keys that are stored in the filesystem. Since the TOE is inaccessible in this situation, administrative zeroization cannot be performed. The keys stored in filesystem are not directly accessible to any user or administrator.**

	Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.2.5 FCS_COP.1/DataEncryption

6.2.5.1 FCS_COP.1/DataEncryption Guidance 1

Objective	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 examined the section titled "**Default Crypto Configuration**" in AGD[1] to verify that it 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. Upon investigation, the evaluator found that AGD[1] states that**:** **The TOE is automatically installed in Default Crypto Configuration. This means that a subset of the cryptographic library functions provided by the TOE are automatically configured to support the values identified in the Security Target.** Specifically: • **Supports the selected modes and key size(s) defined in the Security Target supported by the TOE for data encryption/decryption (FCS_COP.1/DataEncryption).** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.2.6 FCS_COP.1/SigGen

6.2.6.1 FCS_COP.1/SigGen Guidance 1

Objective	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 examined the section titled "**Default Crypto Configuration**" in AGD[1] to verify that it 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. Upon investigation, the evaluator found that AGD[1] states that**:**

	The TOE is automatically installed in Default Crypto Configuration. This means that a subset of the cryptographic library functions provided by the TOE are automatically configured to support the values identified in the Security Target. Specifically**:** • **Supports the selected cryptographic algorithm and key size defined in the Security Target supported by the TOE for signature services (FCS_COP.1/SigGen).** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.2.7 FCS_COP.1/Hash

6.2.7.1 FCS_COP.1/Hash Guidance 1

Objective	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 examined the section titled "**Default Crypto Configuration**" in AGD[1] to verify that it instructs the administrator how to configure the TOE to use the required hash sizes defined in the Security Target. Upon investigation, the evaluator found that AGD[1] states that**:** **The TOE is automatically installed in Default Crypto Configuration. This means that a subset of the cryptographic library functions provided by the TOE are automatically configured to support the values identified in the Security Target.** Specifically**:** • **Supports the selected hash sizes for all cryptographic protocols defined in the Security Target (FCS_COP.1/Hash).** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.2.8 FCS_COP.1/KeyedHash

6.2.8.1 FCS_COP.1/KeyedHash Guidance 1

Objective	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 examined the section titled "**Default Crypto Configuration**" in AGD[1] to verify that it 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. Upon investigation, the evaluator found that AGD[1] states that: **The TOE is automatically installed in Default Crypto Configuration. This means that a subset of the cryptographic library functions provided by the TOE are automatically configured to support the values identified in the Security Target.** Specifically: • **Supports 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 (FCS_COP.1/KeyedHash).** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.2.9 FCS_RBG_EXT.1/ARMA53

6.2.9.1 FCS_RBG_EXT.1 Guidance 1

Objective	The evaluator shall confirm that the guidance documentation contains appropriate instructions for configuring the RNG functionality.
Evaluator Findings	The evaluator examined the section titled "**Default Crypto Configuration**" in AGD[1] to verify that it instructs the administrator how to configure the RNG functionality. Upon investigation, the evaluator found that AGD[1] states that:

	The TOE is automatically installed in Default Crypto Configuration. This means that a subset of the cryptographic library functions provided by the TOE are automatically configured to support the values identified in the Security Target. Specifically**:** • **Supports the RNG functionality specified in the Security Target (FCS_RBG_EXT.1).** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.2.10 FCS_RBG_EXT.1/ARMA72

6.2.10.1 FCS_RBG_EXT.1 Guidance 1

Objective	The evaluator shall confirm that the guidance documentation contains appropriate instructions for configuring the RNG functionality.
Evaluator Findings	The evaluator examined the section titled "**Default Crypto Configuration**" in AGD[1] to verify that it instructs the administrator how to configure the RNG functionality. Upon investigation, the evaluator found that AGD[1] states that**:** **The TOE is automatically installed in Default Crypto Configuration. This means that a subset of the cryptographic library functions provided by the TOE are automatically configured to support the values identified in the Security Target.** Specifically**:** • **Supports the RNG functionality specified in the Security Target (FCS_RBG_EXT.1).** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.2.11 FCS_RBG_EXT.1/Intel

6.2.11.1 FCS_RBG_EXT.1 Guidance 1

Objective	The evaluator shall confirm that the guidance documentation contains appropriate instructions for configuring the RNG functionality.
Evaluator Findings	The evaluator examined the section titled "**Default Crypto Configuration**" in AGD[1] to verify that it instructs the administrator how to configure the RNG functionality. Upon investigation, the evaluator found that AGD[1] states that: **The TOE is automatically installed in Default Crypto Configuration. This means that a subset of the cryptographic library functions provided by the TOE are automatically configured to support the values identified in the Security Target.** Specifically: • **Supports the RNG functionality specified in the Security Target (FCS_RBG_EXT.1).** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.3 Guidance Activities (HTTPS)

6.3.1 *FCS_HTTPS_EXT.1*

6.3.1.1 FCS_HTTPS_EXT.1.1 Guidance 1

Objective	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 ST and determined that the TOE acts as an HTTPS Client communicating with the File Server over TLS. The TOE does not act as an HTTPS Server. The evaluator examined the section titled "**Configuring Communication to the File Server"** in AGD[1] to verify that it instructs the Administrator how to configure TOE for use as an HTTPS client. Upon investigation, the evaluator found that AGD[1] states that:

	The TOE communicates with the File Server using HTTPS over TLS. To perform a file transfer of a new TOE from the File Server perform the following command: • **`software install url <url> tls-service-profile <tls-service-profile name>`** Where: **`<url> = https://<IP address of the File Server>/<filename of the new download>`** **`<tls-service-profile name> =`** the name of the TLS Service Profile. The TLS Service Profile points to the TLS Profile which defines the minimum TLS version, cipher suites, and elliptic curves supported by the TOE and points to the Peer Authentication Profile which defines if the Server's certificate should be validated and if true, how the certificate should be validated. Refer to Section 7 for instructions of how to configure the TLS connection to the File Server in the evaluated configuration. The evaluator examined section 7 titled **Configuring TLS Communication** in AGD[1] to verify that it instructs the Administrator how to configure the TLS parameters for the HTTPS connection. Specifically, section 7 instructs the Administrator how to configure the minimum TLS version, cipher suites, and elliptic curves supported by the TOE and configures the TOE to validate a Server's certificate. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.4 Guidance Activities (NTP)

6.4.1 FCS_NTP_EXT.1

6.4.1.1 FCS_NTP_EXT.1.1 Guidance 1

Objective	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 section titled "**NTP Server Configuration"** in AGD[1] to verify that it provides the 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. Upon investigation, the evaluator found that the AGD states that:
	
	Section "**NTP Server Configuration"** states:
	
	• **By default, the NTP version is v4.**
	
	AGD[1] section "**Configure an NTP Server"** describes how to configure an NTP server.
	
	1. **Configure an association with the NTP server:**
	`system ntp associations remote-ntp-server server-entry <IP address> auth-key-id <integer>`
	2. **Configure message digest algorithm along with message digest string for integrity of time source:**
	`system ntp authentication auth-entry <auth-key-id integer> auth-key-type <sha1>`
	`system ntp authentication auth-entry <auth-key-id integer> auth-key-enc <message digest string>`
	Note: the Ciena device supports both SHA1 and MD5 as the message digest algorithm **however,** the evaluated configuration only supports SHA1.
	3. **Enable NTP message digest authentication:**
	`system ntp authentication auth-admin-state enabled`
	
	To configure multiple NTP servers, repeat the steps above with different IP addresses.
	
	Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass.

6.4.1.2 FCS_NTP_EXT.1.2 Guidance 1

Objective	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.
Evaluator Findings	The evaluator examined the section titled "**NTP Server Configuration"** in AGD[1] 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. Upon investigation, the evaluator found that the AGD[1] states that: 1. **Configure message digest algorithm along with message digest string for integrity of time source:** **`system ntp authentication auth-entry <auth-key-id integer> auth-key-type <sha1>`** **`system ntp authentication auth-entry <auth-key-id integer> auth-key-enc <message digest string>`** Note: the Ciena device supports both SHA1 and MD5 as the message digest algorithm **however,** the evaluated configuration only supports SHA1. **Example** ➢ **`system ntp authentication auth-entry 7 auth-key-type sha1`** ➢ **`system ntp authentication auth-entry 7 auth-key-enc 64b0bfe07a34ab4daf9ab87fda50021e77176151`** a) Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.4.1.3 FCS_NTP_EXT.1.3 Guidance 1

Objective	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 section titled "**NTP Server Configuration"** in AGD[1] to verify that 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. Upon investigation, the evaluator found that the AGD[1] states that: "By default, the TOE does not allow timestamp updates from broadcast and multicast addresses." Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.5 Guidance Activities (SSH)

6.5.1 FCS_SSHS_EXT.1

6.5.1.1 FCS_SSHS_EXT.1.4 Guidance 1

Objective	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 examined the section titled "**Configure Encryption Algorithms**" in AGD[1] to verify that, for each of the selections made in the ST, it instructs the administrator how to configure the TOE to set the set of SSH encryption algorithms. Upon investigation, the evaluator found that the AGD[1] states that:
	Steps
	1. **Configure encryption algorithms.**
	`system ssh-server config encryption-algorithm <encryption-algorithm>`
	Example
	The following command configures the SSH encryption algorithms in the evaluated configuration.
	`system ssh-server config encryption-algorithm aes-128-ctr aes-256-ctr aes128-gcm@openssh.com aes256-gcm@openssh.com`
	Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass.

6.5.1.2 FCS_SSHS_EXT.1.5 Guidance 1

Objective	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 examined the section titled "**Configure the PKA Authentication Implementation**" in AGD[1] to verify that, for each of the selections made in the ST, it instructs the administrator how to configure the TOE to set the set of SSH PKA algorithms. Upon investigation, the evaluator found that the AGD[1] states that: **Steps** 1. **Configure encryption algorithms.** **system ssh-server config pka-algorithm <pka-algorithm>** Example **The following command configures the SSH encryption algorithms in the evaluated configuration.** **system ssh-server config pka-algorithm ssh-rsa ecdsa-sha2-nistp256** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.5.1.3 FCS_SSHS_EXT.1.6 Guidance 1

Objective	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 examined the section titled "**Configure MAC algorithms**" in AGD[1] to verify that, for each of the selections made in the ST, it instructs the administrator how to configure the TOE to set the set of MAC algorithms. Upon investigation, the evaluator found that the AGD[1] states that: Steps 1. Configure encryption algorithms. system ssh-server config mac-algorithm <mac-algorithm> Example The following command configures the SSH encryption algorithms in the evaluated configuration. system ssh-server config mac-algorithm hmac-sha2-256-etm hmac-sha2-512-etm hmac-sha1

	Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.5.1.4 FCS_SSHS_EXT.1.7 Guidance 1

Objective	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 examined the section titled "**Configure Key exchange Algorithms**" in AGD to verify that, for each of the selections made in the ST, it instructs the administrator how to configure the TOE to set the set of key exchange algorithms. Upon investigation, the evaluator found that the AGD[1] states that:
	Steps
	1. **Configure key exchange algorithms.**
	`System ssh-server config kex-algorithm <kex-algorithm>`
	Example
	The following command configures the SSH key exchange algorithms in the evaluated configuration.
	`System ssh-server config kex-algorithm ecdh-sha2-nistp256 ecdh-sha2-nistp384 ecdh-sha2-nistp521 diffie-hellman-group14-sha1 diffie-hellman-group14-sha256 diffie-hellman-group16-sha512`

Verdict	Pass.

6.5.1.5 FCS_SSHS_EXT.1.8 Guidance 1

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

Evaluator Findings	The evaluator examined the section titled "**Configure the Rekey Time**" in AGD[1] to verify that, for each of the selections made in the ST, it instructs the administrator how to configure the TOE to set the rekey time. Upon investigation, the evaluator found that the AGD[1] states that:
	Steps
	1 Configure the rekey time.
	`System ssh-server config rekey-time <rekey-time>`
	Example
	The following command configures the SSH rekey time to the evaluated configuration.
	`system ssh-server config rekey-time 3600`
	Based on these findings, this assurance activity is considered satisfied.
	The evaluator examined the section titled "Configure the Rekey Limit" in AGD[1] to verify that, for each of the selections made in the ST, it instructs the administrator how to configure the TOE to set the rekey limit. Upon investigation, the evaluator found that the AGD[1] states that:
	Steps
	1 Configure the rekey limit.
	`System ssh-server config rekey-limit <rekey-limit>`
	Example
	The following command configures the SSH rekey limit to the evaluated configuration.
	`system ssh-server config rekey-limit 1G`
	Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass.

6.6 Guidance Activities (TLS)

6.6.1 FCS_TLSC_EXT.1

6.6.1.1 FCS_TLSC_EXT.1.1 Guidance 1

Objective	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 examined the ST's TSS section and AGD[2] and determined the TOE's supported cipher suites is a smaller list than the TOE's default supported suites and that the default minimum TLS version of the TOE is 1.1 and the ST supports 1.2 and rejects all other TLS versions.
	
	The evaluator then examined the section titled "**Configuring TLS Communication**" in AGD[1] to verify that it provides instructions for configuring the supported TLS version and supported cipher suites. Upon investigation, the evaluator found that AGD[1] states that:
	
	A TLS Profile defines the minimum TLS version, cipher suites, elliptic curves, and session timeout value for a TLS connection. A TLS Profile must be configured for the evaluated configuration because the default values of the TLS version, cipher suites, and elliptic curves are not supported by the TOE.
	
	Upon investigation, the evaluator found that AGD[1] describes the tls-version parameter used to define the minimum TLS version and states that it Sets the minimum TLS version. This parameter must be configured to tls-1.2. The guidance further gives instructions for populating the TLS Profile's tls-version parameter:
	
	1. Set the minimum TLS version for **the profile named syslog-tls:**
	`diag@CGSI5162# hello-params syslog-tls tls-versions tls-version tls-1.2`
	
	Upon further investigation, the evaluator found that AGD[1] describes the cipher-suite parameter used to define the cipher suites and states that it Identifies the cipher suites to use for the profile and This parameter must be configured to the cipher suites defined in the ST. The guidance further gives the instructions for populating the TLS Profile's cipher-suite parameter:
	
	2. Set the TLS cipher suites for the profile named syslog-tls:
	`hello-params syslog-tls cipher-suites cipher-suite TLS_RSA_WITH_AES_128_CBC_SHA`
	`hello-params syslog-tls cipher-suites cipher-suite TLS_RSA_WITH_AES_256_CBC_SHA`

	hello-params syslog-tls cipher-suites cipher-suite TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA **hello-params syslog-tls cipher-suites cipher-suite TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA** **hello-params syslog-tls cipher-suites cipher-suite TLS_RSA_WITH_AES_128_CBC_SHA256** **hello-params syslog-tls cipher-suites cipher-suite TLS_RSA_WITH_AES_256_CBC_ SHA256** **hello-params syslog-tls cipher-suites cipher-suite TLS_RSA_WITH_AES_128_GCM_SHA256** **hello-params syslog-tls cipher-suites cipher-suite TLS_RSA_WITH_AES_256_GCM_SHA384** **hello-params syslog-tls cipher-suites cipher-suite TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256** **hello-params syslog-tls cipher-suites cipher-suite TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384** **hello-params syslog-tls cipher-suites cipher-suite TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256** **hello-params syslog-tls cipher-suites cipher-suite TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.6.1.2 FCS_TLSC_EXT.1.2 Guidance 1

Objective	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 examined the section titled "**X509 Certificates** " in AGD[1] to verify that it describes all supported identifiers, explicitly states whether the TOE supports the SAN extension or not, includes detailed instructions on how to configure the reference identifier(s) used to check the identity of peer(s), and provides a set of warnings and/or CA policy recommendations that would result in secure TOE use. Upon investigation, the evaluator found that the AGD states that: **By default, the TOE supports SAN extension and checks SAN extension over CN when present. The TOE ignores CN when SAN is present. When SAN is not present, the TOE falls back to CN check. FQDN is supported in both SAN and CN while IP address is only supported in SAN.** **By default, the TOE supports wildcards in certificates. The wildcard must be in the left-most label of the presented identifier and can only cover one level of subdomains. For the reference identifier without a left-most label as in the**

	certificate, the connection will fail, i.e., awesome.com doesn't match *.awesome.com.s. The TLS client does not support certificate pinning. The evaluator examined the section titled "Create a Peer Authentication Profile", "Creating a TLS service profile", and "Configuring Syslog" in AGD[1] to verify that it includes detailed instructions on how to configure the reference identifier(s) used to check the identity of peer(s). Upon investigation, the evaluator found that the AGD states that: AGD Section: Create a Peer Authentication Profile • **Specify the name of the peer authentication profile:** `pkix peer-auth-profiles peer-auth-profile <peer-auth-profile-name>` • **Set the check for expiration is performed on the Server's X.509 certificate. This must be set to true for the evaluated configuration.** `pkix peer-auth-profiles peer-auth-profile <peer-auth-profile-name> check-cert-expiry true` AGD Section: Creating a TLS service profile • **Identify the TLS Profile** `tls-service-profiles <tls-service-profile-name> tls-profile-name <tls-profile>` • **Identify the peer authentication profile.** `tls-service-profiles <tls-service-profile-name> tls-peer-auth-profile-name <peer-auth-profile>` • **Identify the TLS certificate.** `tls-service-profiles <tls-service-profile-name> tls-certificate-name <certificate>` AGD Section: Configuring Syslog • **Identify the IP address of the syslog server.** `Config syslog log-actions remote-syslog-tls destination <ip address or DNS>` • **Assign a TLS Profile to the connection.**

	`syslog log-actions remote-syslog-tls tls-service-profile "profile name"` Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.6.1.3 FCS_TLSC_EXT.1.4 Guidance 1

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

Evaluator Findings	The evaluator examined the ST's TSS section and determined that Supported Elliptic Curves must be configured to meet the SFR. The evaluator then examined the section titled "**Configuring TLS Communication**" in AGD[1] to verify that it provides instructions for configuring the supported elliptic curves. . Upon investigation, the evaluator found that AGD[1] states that:
	
	A TLS Profile defines the TLS version, cipher suites, elliptic curves, and session timeout value for a TLS connection. A TLS Profile must be configured for the evaluated configuration because the default values of the TLS version, cipher suites, and elliptic curves are not supported by the TOE.
	
	Upon investigation, the evaluator found that AGD[1] describes the elliptic-curve parameter used to define the supported elliptic curves and states that This parameter must be configured to the elliptic curves defined in the ST. The guidance further gives the instructions for populating the TLS Profile's elliptic curves parameters:
	
	3. Set the elliptic curves for the profile named syslog-tls:
	`hello-params syslog-tls elliptic-curves elliptic-curve secp256r1`
	`hello-params syslog-tls elliptic-curves elliptic-curve secp384r1`
	`hello-params syslog-tls elliptic-curves elliptic-curve secp521r1`
	
	Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass.

6.7 Guidance Activities (Identification and Authentication)

6.7.1 FIA_AFL.1

6.7.1.1 FIA_AFL.1 Guidance 1

Objective	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 section titled "**User Lockout Policy**" in AGD[1] to verify that it provides instructions for configuring the number of successive unsuccessful authentication attempts and time period (if implemented), 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). Upon investigation, the evaluator found that AGD[1] states that: **The evaluated configuration requires the Administrator to configure the user lockout-policy to set a level of sequential login failures to lock the account until the lockout period has expired. Only SSH protocol is supported for configuration of user lockout attributes.** **Note: Administrator lockouts are not applicable to the local console. Local administrators cannot be locked out and have the ability to unlock other users by using the local console.** **Steps** • **Configure the number of failed login attempts before lockout:** system aaa authentication lockout-policy config fail-limit <integer> Note: The configurable integer is in the range 1-5. • **Configure the duration of the lockout:** system aaa authentication lockout-policy config lockout-time<integer> Note: The value in integer is for seconds as unit. **Example** **The following commands lock out the user for one minute after three failed login attempts.**

	`system aaa authentication lockout-policy config fail-limit 3`
	`system aaa authentication lockout-policy config lockout-time 60`
	
	Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass.

6.7.1.2 FIA_AFL.1 Guidance 2

Objective	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 section titled "**User Lockout Policy**" in AGD[1] to verify 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. Upon investigation, the evaluator found that AGD[1] states that: **Note: Administrator lockouts are not applicable to the local console. Local administrators cannot be locked out and has the ability to unlock other users by using the local console.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.7.2 FIA_PMG_EXT.1

6.7.2.1 FIA_PMG_EXT.1.1 Guidance 1

Objective	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 section titled "**Passwords Rules**" in AGD[1] to verify that it identifies the characters that may be used in passwords and provides guidance to security administrators on the composition of strong passwords and

	provides instructions on setting the minimum password length and describes the valid minimum password lengths supported. Upon investigation, the evaluator found that AGD[1] states that:
	
	The following commands provide an example of implementing a company's password-policy rules. Consult your company's individual password-policy rules before configuring the password policy commands. ***Note***: For more information about password policy commands refer to your Ciena platform's Administration SAOS 10.7.1 Guide.
	
	Note: to configure the TOE in the evaluated configuration the Administrator must set the minimum length of the passwords.
	
	Steps
	
	1. Configure the minimum length of the password:
	`system aaa authentication password-policy config minlength <integer>`
	
	2. Configure the minimum number of lower-case characters:
	
	`system aaa authentication password-policy config minlowercase-chars <integer>`
	
	3. Configure the minimum number of numeric characters:
	`system aaa authentication password-policy config minnumeric-chars <integer>`
	
	4. Configure the minimum number of special characters:
	`system aaa authentication password-policy config minspecial-chars <integer>`
	
	5. Configure the minimum number of upper-case characters:
	`system aaa authentication password-policy config minuppercase-chars <integer>`
	
	Example
	
	The following example configures the user password-policy. In this example, the password may not contain dictionary words, username or its reverse. It also requires that the password be at least 10 characters long and contain at least one lowercase character and one numeric character. It does not require the password to contain any special characters.
	
	`system aaa authentication password-policy config disallow-dict-words on`

	system aaa authentication password-policy config disallow-username on system aaa authentication password-policy config min-length 10 system aaa authentication password-policy config min-lowercase-chars 1 system aaa authentication password-policy config min-numeric-chars 1 system aaa authentication password-policy config minspecial-chars 0 Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.7.3 FIA_UIA_EXT.1

6.7.3.1 FIA_UIA_EXT.1 Guidance 1

Objective	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 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.
Evaluator Findings	The evaluator examined the operational guidance to determine that any necessary preparatory steps to logging in are described. Several relevant sections were used to determine the verdict of this assurance activity. The evaluator found that the AGD provides instructions for configuring user authentication on the TOE in the following sections: • **Section "Accessing the TOE"** describes how to physically connect the console and LAN administrator interfaces**.** • **Section "Configuring the Remote Management Interface"** describes the procedures to configure SSHv2 for remote access. • **Section "User Account Configuration and Management"** describes how to configure users and their roles. • **Section "Login Banners"** describes that no configuration is necessary to limit the services prior to login. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.7.4 FIA_UAU.7

6.7.4.1 FIA_UAU.7 Guidance 1

Objective	The evaluator shall examine AGD[1] 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 section titled 'Protection Authentication Feedback' in AGD[1] to verify that it describes any necessary preparatory steps to ensure authentication data is not revealed while entering for each local login allowed. Upon investigation, the evaluator found that AGD[1] states that: **The TOE does not provide any feedback for the password characters entered. This is by default and does not require any configuration.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.7.5 FIA_X509_EXT.1/Rev

6.7.5.1 FIA_X509_EXT.1/Rev Guidance 1

Objective	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 examined the section titled "**X.509 Certificates**" in AGD[1] to verify that it 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 describe how certificate revocation checking is performed. Upon investigation, the evaluator found that AGD[1] states that: **Certificates are validated upon receipt from the server (Syslog or File Server) and when they are loaded onto the TOE. Certificate validity is checked on each certificate validation. If the validation of the certificate fails because the OCSP Server cannot be connected to, the certificate shall not be accepted.** 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 will only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.**
	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.
	Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass.

6.7.6 FIA_X509_EXT.2

6.7.6.1 FIA_X509_EXT.2 Guidance 1

Objective	The evaluator shall check the administrative guidance to ensure that it includes any necessary instructions for configuring the operating environment so that the TOE can use the certificates.

Evaluator Findings	The evaluator examined the section titled "**OCSP Server Requirements**" in AGD[1] to ensure that it includes any necessary instructions for configuring the operating environment so that the TOE can use the certificates. Upon investigation, the evaluator found that AGD[1] states that: **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.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.7.6.2 FIA_X509_EXT.2 Guidance 2

Objective	The guidance documentation shall also include any required configuration on the TOE to use the certificates.
Evaluator Findings	The evaluator examined the section titled "Configure the Certificates Required for the TOE" in AGD[1] to ensure that it includes any necessary instructions for configuring the operating environment so that the TOE can use the certificates. Upon investigation, the evaluator found that AGD[1] states that: **To configure the certificates required for the TOE, perform the following steps.** ➢ **Install a CA certificate:** **pkix-ca install ca-cert-name <ca_cert_name> remote-fileuri scp://<server_ip>/<cert_path>/<ca.cert> login-id <login_id> password <login_password>** **The following example installs a CA certificate named test.** **pkix-ca install ca-cert-name test remote-file-uri scp://192.0.2.0/certs/ SaosCertificate.pem login-id User1 password abc** ➢ **Install a device certificate and private key as required by the network plan.** **Steps**

	Install a device certificate and private key:
	
	`pkix-certificates install <cert_name> remote-file-uri scp://<server_ip>/<cert_path>/<device.p12> login-id <login_id> password <login_password> cert-passphrase <cert_pass_phrase>`
	
	The following example installs a device certificate and private key.
	
	`pkix-certificates install TestCa remote-file-uri scp://192.0.2.0/certs/ TestClient.p12 login-id User1 password abc cert-passphrase test`
	
	➢ **Generate a private key and certificate signing request on the system, sign the certificate externally, and install the certificate as required.**
	Steps
	
	1 Generate a private key and certificate signing request on the system, sign the
	
	certificate externally, and install the certificate:
	
	`pkix-certificates-csr-generate cert-name <cert_name> algorithm-identifier <algorithm-identifier> remote-fileuri ftp://server_ip/<path>/<cert.cnf> cert-passphrase <cert_passPhrase>`
	
	The following example generates a private key and certificate signing request.
	
	`pkix-certificates-csr-generate cert-name testCsrGen algorithm-identifier`
	
	`pkix-types:rsa1024 remote-file-uri ftp://1.2.3.4/certs/ClientCert.pem certpassphrase test`
	
	➢ **Enable check-fingerprint:**
	pkix peer-auth-profiles peer-auth-profile <peer-authprofile> check-fingerprint <true|false>

	The following example enables check-fingerprint for a peer authentication: **profile named baseConf.** **`pkix peer-auth-profiles peer-auth-profile baseConf check-fingerprint true`** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass/Fail

6.7.6.3 FIA_X509_EXT.2 Guidance 3

Objective	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 examined the section titled "**X.509 Certificates"** in AGD[1] to verify that the guidance documentation 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. Upon investigation, the evaluator found that the TSS states that: **If the validation of the certificate fails because the OCSP Server cannot be connected to, the certificate shall not be accepted. If the connection fails, the Administrator should check the physical connections and reenable the OCSP client with the following command: `hello-params baseConf ocsp-state enabled`. Where `baseConf` is the TLS Profile for the OCSP Server.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass/Fail

6.8 Guidance Activities (Security Management)

6.8.1 FMT_MOF.1/ManualUpdate

6.8.1.1 FMT_MOF.1/ManualUpdate Guidance 1

Objective	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 section titled "**Updating the TOE"** in the Security Target to verify that the guidance documentation describes the necessary steps to perform a manual updated. Upon investigation, the evaluator found that the TSS states that: The TOE can be updated from the File Server using HTTPS over TLS. The TOE verifies the update using signature verification. If the signature validation is successful, the TOE will be immediately applied. If the signature validation fails, an error message will be displayed. Steps: • `software install url <url> tls-service-profile <tls-service-profile name>` Where: `<url> = https://<IP address of the File Server>/<filename of the new download>` `<tls-service-profile name> =` the name of the TLS Service Profile. The TLS Service profile points to the TLS Profile which defines the minimum TLS version, cipher suites and elliptic curves supported by the TOE and points to the Peer Authentication Profile which defines if the Server's certificate should be validated and if true, how the certificate should be validated. Refer to Section 6 of the AGD document for instructions of how to configure the TLS connection to the File Server in the evaluated configuration. The following is an example of installing an image with new download named saos-10-07-01-0289-RS12.yml from the file server with IP address of 10.1.5.208. The TLS Service Profile's name is syslog-tls-service. 

	Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass/fail

6.8.2 FMT_MOF.1/Functions Management of security functions behaviour

6.8.2.1 FMT_MOF.1/Functions Guidance 2

Objective	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 ST and determined the FMT_MOF.1/Functions claimed how to enable syslog. The evaluator then examined section "Configuring Syslog" of AGD[1] and determined that to enable syslog the following commands are required: **`Config syslog log-actions remote-syslog-tls destination <ip address or DNS>`** **`syslog log-actions remote-syslog-tls admin-state <enable/disable>`** **Based on these findings, this assurance activity is considered satisfied.**
Verdict	Pass/fail

6.8.3 FMT_MTD.1/CoreData Management of TSF Data

6.8.3.1 FMT_MTD.1/CoreData Guidance 1

Objective	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 cPP is identified, and that configuration information is provided to ensure that only administrators have access to the functions.
Evaluator Findings	The evaluator examined the section titled "**User Account Configuration and Management**" in AGD[1] to verify that it identifies each of the TSF-data-manipulating functions implemented in response to the requirements of the cPP. Upon investigation, the evaluator found that the AGD states that:

	The TOE requires successful identification and authentication of each administrator prior to granting them access to the TOE. Access to the TOE is by making available to the user a shell in which the user can execute CLI commands. Without access to the shell, the CLI is not accessible to the user and, consequently, administrator accesses are not possible. There are no management functions other than those accessible through the CLI. **The only access the TOE allows prior to the successful identification and authentication of the user is the access banner displayed at each login prompt.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.8.3.2 FMT_MTD.1/CoreData Guidance 2

Objective	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	The evaluator examined the section titled "**Configure the Certificates Required for the TOE**" in AGD[1] to ensure that it includes any necessary instructions for configuring the operating environment so that the TOE can use the certificates. Upon investigation, the evaluator found that AGD[1] states that: **To configure the certificates required for the TOE, perform the following steps.** ➢ **Install a CA certificate:** **pkix-ca install ca-cert-name <ca_cert_name> remote-fileuri scp://<server_ip>/<cert_path>/<ca.cert> login-id <login_id> password <login_password>** **The following example installs a CA certificate named test.**

	pkix-ca install ca-cert-name test remote-file-uri scp://192.0.2.0/certs/ SaosCertificate.pem login-id User1 password abc ➢ **Install a device certificate and private key as required by the network plan.** **Steps** **Install a device certificate and private key:** pkix-certificates install <cert_name> remote-file-uri scp://<server_ip>/<cert_path>/<device.p12> login-id <login_id> password <login_password> cert-passphrase <cert_pass_phrase> **The following example installs a device certificate and private key.** pkix-certificates install TestCa remote-file-uri scp://192.0.2.0/certs/ TestClient.p12 login-id User1 password abc cert-passphrase test ➢ **Generate a private key and certificate signing request on the system, sign the certificate externally, and install the certificate as required.** **Steps** **1 Generate a private key and certificate signing request on the system, sign the** **certificate externally, and install the certificate:** pkix-certificates-csr-generate cert-name <cert_name> algorithm-identifier <algorithm-identifier> remote-fileuri ftp://server_ip/<path>/<cert.cnf> cert-passphrase <cert_passPhrase> **The following example generates a private key and certificate signing request.** pkix-certificates-csr-generate cert-name testCsrGen algorithm-identifier

	pkix-types:rsa1024 remote-file-uri ftp://1.2.3.4/certs/ClientCert.pem certpassphrase test ➢ **Enable check-fingerprint:** **pkix peer-auth-profiles peer-auth-profile <peer-authprofile> check-fingerprint <true|false>** **The following example enables check-fingerprint for a peer authentication** **profile named baseConf.** **pkix peer-auth-profiles peer-auth-profile baseConf check-fingerprint true** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.8.4 FMT_MTD.1/CryptoKeys Management of TSF Data

6.8.4.1 FMT_MTD.1/CryptoKeys Guidance 2

Objective	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.
Evaluator Findings	The evaluator examined the section titled "**SSH Public Key Configuration**" in the AGD to verify that 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. Upon investigation, the evaluator found that the AGD states that: **system ssh-server user-pubkey install user-name <user> filename <filename> [server-type <server-type>] address** **<address> [login <login-id> password <password>]** **OR**

	system ssh-server user-pubkey install user-name <user> url <url>
	
	The evaluator examined the section titled "**Creating a TLS service profile**" in the AGD to verify that 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. Upon investigation, the evaluator found that the AGD states that:
	
	1. **Identify the TLS Profile**
	`tls-service-profiles <tls-service-profile-name> tls-profile-name <tls-profile>`
	2. **Identify the peer authentication profile.**
	`tls-service-profiles <tls-service-profile-name> tls-peer-auth-profile-name <peer-auth-profile>`
	3. **Identify the TLS certificate.**
	`tls-service-profiles <tls-service-profile-name> tls-certificate-name <certificate>`
	
	Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass.

6.8.5 FMT_SMF.1 Specification of Management Functions

6.8.5.1 FMT_SMF.1 Guidance 1

Objective	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. The evaluator shall confirm that the TSS details which security management functions are available through which interface(s) (local administration interface, remote administration interface). The evaluator shall examine the TSS and Guidance Documentation to verify they both describe the local administrative interface. The evaluator shall ensure the AGD includes appropriate warnings for the administrator to ensure the interface is local.

Evaluator Findings

The evaluator examined the TSS FMT_SMF.1 section of the ST to verify that it details which security management functions are available and examined AGD[1] and determined the management functions are described in the following table:

Management Functions Defined in the FMT_SMF.1	SFR	AGD[1] Section Name
Ability to administer the TOE locally and remotely.	FIA_UIA_EXT.1	**Accessing the TOE, Configuring the Remote Management Interface, User Account Configuration and Management**
Ability to configure the access banner.	FTA_TAB.1	**Login Banners**
Ability to configure the session inactivity time before session termination or locking.	FTA_SSL.3	**SSH Ilde Session Termination**
Ability to update the TOE, and to verify the updates using digital signature and [no other] capability prior to installing those updates.	FPT_TUD_EXT.1, FMT_MOF.1/ManualUpdate	**Updating the TOE**
Ability to configure the authentication failure parameters for FIA_AFL.1.	FIA_AFL.1, FIA_AFL.1	**User Lockout**
Ability to modify the behaviour of the transmission of audit data to an external IT entity.	FAU_STG_EXT.1,	**Configuring Syslog, Configuring Log Level**
Ability to manage the cryptographic keys.	FMT_MTD.1/CryptoKeys	**Configure the Certificates Required for the TOE**
Ability to configure thresholds for SSH rekeying.	FCS_SSHS_EXT.1, FCS_SSHS_EXT.1	**Configure the Rekey Time**
Ability to set the time which is used for time-stamps.	FPT_STM.EXT.1	**Clock Management**

	Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors.	FIA_X509_EXT.1.1/Rev	**Configure the Certificates Required for the TOE**
	Ability to import x509v3 certificates on TOE's trust store.	FIA_X509_EXT.1.1	**X.509 Certificates**
	Ability to configure NTP.	FCS_NTP_EXT.1	**Configure an NTP Server**
Verdict	Pass.		

6.8.6 FMT_SMR.2 Restrictions on Security Roles

6.8.6.1 FMT_SMR.2 Guidance 1

Objective	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 examined the section titled "**Accessing the TOE**" in AGD[1] to verify 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. Upon investigation, the evaluator found that AGD[1] includes the following information. **System access to the system can be established by means of:** • **console port. The console port is used to access the system by means of a laptop PC. The serial console port is a Serial EIA-561 (RJ-45) or USBC port. The console port allows for local CLI access to the system (Section 4.1).** • **Secure Shell (SSH). SSH provides remote login for remote CLI access to the system and perform SFTP file transfers. SSH verifies and grants access to login requests by encrypting user ID and passwords or through public key encryption. SSH/SFTP is supported over IPv4 (Section 5).** The evaluator found that Section 4.1 describes how to physically connect to the console port and includes and how to login into the system. The evaluator found that Section 5 describes configuration of the SSHv2 management interface including a description of all of the required parameters and their values.

	The evaluator found that Section Configure Remote Interface and Administration Protocols states: **Up to two remote management interfaces are permitted. Each client terminal requires SSHv2 Client software installed and running.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

6.9 Guidance Activities (Protection of the TSF)

6.9.1 FPT_STM_EXT.1 Reliable Time Stamps

6.9.1.1 FPT_STM_EXT.1 Guidance 1 [TD0632]

Objective	The evaluator examines the guidance documentation to ensure it instructs the administrator how to set the time. 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. 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 evaluator examined the section titled "**Clock Management"** in AGD[1] to verify that it instructs the administrator how to set the time. Upon investigation, the evaluator found that the AGD states that: **1 Set the system time:** **system set clock <current datetime>** **Example** **The following example sets the system time to 2019-08-21T22:25:00Z.** **system set clock 2019-08-21T22:25:00Z**

	The evaluator examined section titled "**NTP Server Configuration"** in the AGD to verify that it instructs the administrator how to set the time. Upon investigation, the evaluator found that the AGD states that: **1. Create the NTP association:** **system ntp associations remote-ntp-server server-entry <IP address> admin-state enabled.** **2. Enable the NTP client:** **system ntp admin-state enabled.** **3. Configure an association with the NTP server:** **system ntp associations remote-ntp-server server-entry <IP address> auth-key-id <integer>** **4. Configure message digest algorithm along with message digest string for integrity of time source:** **system ntp authentication auth-entry <auth-key-id integer> auth-key-type <sha1/md5>** **system ntp authentication auth-entry <auth-key-id integer> auth-key-enc <message digest string>** **5. Enable NTP message digest authentication:** **system ntp authentication auth-admin-state enabled** **Example** **system ntp associations remote-ntp-server server-entry 192.0.2.0 admin-state enabled** **system ntp associations remote-ntp-server server-entry 192.0.2.0 auth-key-id 7** **system ntp admin-state enabled** **system ntp authentication auth-entry 7 auth-key-type sha1** **system ntp authentication auth-entry 7 auth-key-enc 64b0bfe07a34ab4daf9ab87fda50021e77176151**

	system ntp authentication auth-admin-state enabled show ntp client +-------- NTP CLIENT STATE -------+ | Name | Value | +----------------------+----------+ | Admin State | enabled | | Mode | polling | | Polling Min Interval | 16 | | Polling Max Interval | 16 | | Auth Admin State | disabled | | Synchronized | True | | Delay | 0.124 | | Offset | -2.213 | | Jitter | 5.919 | | Drift (PPM) | 0.0 | +----------------------+----------+ +---------- NTP CONFIGURED SERVERS ---------+ | Address | Auth Key ID | Admin State | +---------------+-------------+-------------+

	| 192.0.2.0 | 7 | enabled | **+---------------+-------------+-------------+** **+----------------------- NTP OPER SERVERS ----------------------------+** **| Address | Auth | Server | Server Condition | Auth State| Offset |** **| |Key ID | State | | | |** **+---------------+-------+--------+-----------------+-----------+--------+** **| 192.0.2.0 | 7 | reach | syspeer | none | -2.213 |** **+---------------+-------+--------+-----------------+-----------+--------+** The selection "obtain time from the underlying virtualization system" is not selected in ST, therefore its corresponding assurance activity is not applicable. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass

6.9.2 FPT_TST_EXT.1.1 TSF testing

6.9.2.1 FPT_TST_EXT.1.1 Guidance 1

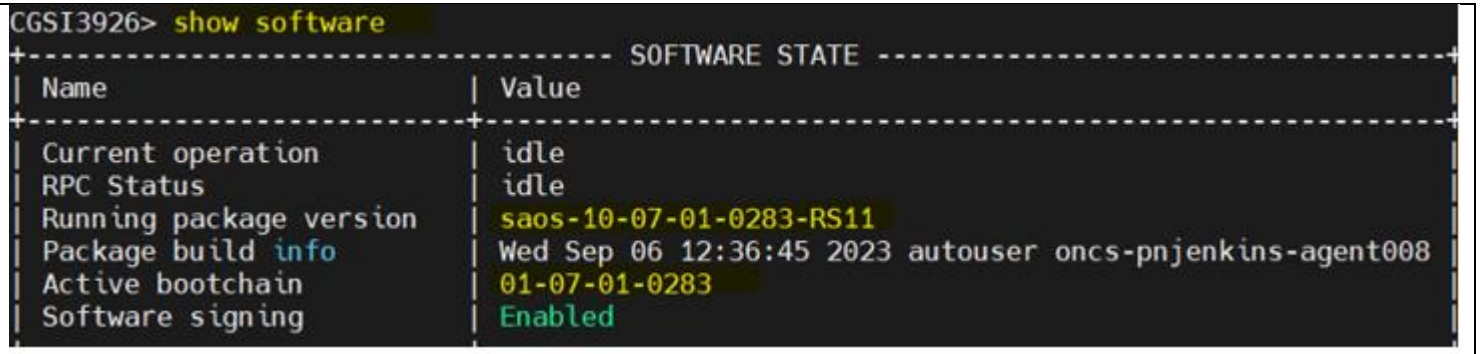
Objective	The evaluator shall also ensure that the guidance documentation describes the possible errors that may result from such tests, and actions the administrator should take in response; these possible errors shall correspond to those described in the TSS.

Evaluator Findings	The evaluator examined the section titled "**Self-Tests"** in the AGD to verify that it describes the possible errors that may result from such tests, and actions the administrator should take in response. Upon investigation, the evaluator found that the AGD states that: **If the self-tests fail, the failure will be reported on the workstation's screen. If any self-test fails, the Administrator should contact Ciena support at www.ciena.com.** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.9.3 FPT_TUD_EXT.1 Trusted Update

6.9.3.1 FPT_TUD_EXT.1 Guidance 1

Objective	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 examined the section titled "**Verifying the TOE Version**" in AGD[1] to verify that it describes how to query the currently active version. The evaluator also examined the section titled "**Updating the TOE**" and determined that the TOE does not support delayed activation of new uploaded versions of the TOE. Upon investigation, the evaluator found that AGD[1] states that: **To verify the current version of the TOE perform the following command:** • **`show software`** The following displays an example output of the show software command.

	 Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.9.3.2 FPT_TUD_EXT.1 Guidance 2

Objective	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 examined the TSS FPT_TUD_EXT.1 section of the ST and verified that product updates are validated using a SHA-256 digital signature. The evaluator examined the section titled "**Secure Acceptance of the TOE**" in AGD[1] to verify that it describes how the verification of the authenticity of the update is performed. Upon investigation, the evaluator found that AGD[1] states that: **When the TOE is updated using HTTPS over TLS, the TOE image is validated with a SHA-256 digital signature. The TOE will display messages to the workstation indicating the success and or failure of the signature verification.** The evaluator also examined the section titled "**Successful Upload Signature Verification**" in AGD[1] to verify that it describes the procedures for successful verification. Upon investigation, the evaluator found that AGD[1] states that: **Note that once the "software install" command has been issued, and if the signature validation is successful, there is no other administrative action required. The TOE will automatically install and activate the new image.**

	The guidance included the console displays of a successful installation to guide the Administrator on detecting a successful installation. The evaluator also examined the section titled "**Unsuccessful Upload Signature Verification**" in AGD[1] to verify that it describes the procedures for an unsuccessful verification. Upon investigation, the evaluator found that AGD[1] states that: If an upgrade fails, go to the following website to report the error: https//www.ciena.com. The guidance included the console display of an unsuccessful installation to guide the Administrator on detecting an unsuccessful installation. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.9.3.3 FPT_TUD_EXT.1 Guidance 3

Objective	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	N/A. Published hash verification is not supported by the TOE.
Verdict	N/A Published hash verification is not supported by the TOE.

6.9.3.4 FPT_TUD_EXT.1 Guidance 6

Objective	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 update mechanism is not supported by the TOE.**
Verdict	N/A. Certificate-based update mechanism is not supported by the TOE.

6.10 Guidance Activities (TOE Access)

6.10.1 FTA_SSL_EXT.1 TSF-Initiated Session Locking

6.10.1.1 FTA_SSL_EXT.1 Guidance 1

Objective	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 examined the section titled "**SSH Idle Session Termination**" in AGD[1] to verify whether local administrative session locking or termination is supported and instructions for configuring the inactivity time period. Upon investigation, the evaluator found that the AGD states that: **The evaluated configuration requires the Administrator to set a session termination configuration for an SSH session that has been inactive for an Administrative configurable amount of time. This configuration will apply to both the console and remote administrative logins. To set the SSH idle timeout perform the following command.** **Steps** **1. Set the SSH idle timeout:** ➢ `system ssh-server config timeout <1-65535>` **Example** **The following example command sets the SSH idle timeout to one minute, that is, 60 seconds.** ➢ `system ssh-server config timeout 60` Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.10.2 FTA_SSL.3 TSF-Initiated Termination

6.10.2.1 FTA_SSL.3 Guidance 1

Objective	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 examined the section titled "**SSH Idle Session Termination**" in AGD[1] to verify that it includes instructions for configuring the inactivity time period for remote administrative session termination. Upon investigation, the evaluator found that AGD[1] states that: **Steps** **2. Set the SSH idle timeout:** ➢ **`system ssh-server config timeout <1-65535>`** **Example** The following example command sets the SSH idle timeout to one minute, that is, 60 seconds. ➢ `system ssh-server config timeout 60` Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.10.3 FTA_SSL.4 User-Initiated Termination

6.10.3.1 FTA_SSL.4 Guidance 1

Objective	The evaluator shall confirm that the guidance documentation states how to terminate a local or remote interactive session.
Evaluator Findings	The evaluator examined the section titled "**User Session Termination**" in AGD[1] to verify that it states how to terminate a local or remote interactive session. Upon investigation, the evaluator found that the AGD states that: **The TOE allows termination of the user's own interactive session using the 'exit' command. This command applies to both local and remote sessions.**

	Example • **exit** Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.10.4 FTA_TAB.1 Default TOE Access Banners

6.10.4.1 FTA_TAB.1 Guidance 1

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

Evaluator Findings	The evaluator examined the section titled "**Login Banners**" in AGD[1] to verify that it describes how to configure the banner message. Upon investigation, the evaluator found that the AGD states that:
	To create a banner of text "This is a banner" use the command.
	Steps
	1. Set the system welcome-banner:
	`system config motd-banner <banner-text>`
	Example
	The following example sets the system welcome-banner:
	`system config motd-banner "This is a banner"`
	Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass.

6.11 Guidance Activities (Trusted Path/Channels)

6.11.1 FTP_ITC.1 Inter-TSF Trusted Channel

6.11.1.1 FTP_ITC.1 Guidance 1

Objective	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 examined the TSS FPT_ITC.1 section of the ST and determined that the TOE supports: • TLS communication with the audit server and • HTTPS over TLS communication with the file server. Refer to "FAU_STG_EXT.1 Guidance 1" of this document for the evaluation of configuring syslog. Refer to "FCS_HTTPS_EXT.1.1 Guidance 1" of this document for the evaluation of configuring the file server. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

6.11.2 FTP_TRP.1/Admin Trusted Path

6.11.2.1 FTP_TRP.1/Admin Guidance 1

Objective	The evaluator shall confirm that the guidance documentation contains instructions for establishing the remote administrative sessions for each supported method.
Evaluator Findings	The evaluator examined the section titled "**Configuring the Remote Management Interface (SSHv2)**" and "**SSH Public Key Configuration**" in AGD[1] to verify that it contains instructions for establishing the remote administrative sessions for each supported method. Upon investigation, the evaluator found that the AGD states that: **The TOE only implements SSHv2 to support remote client administrative management.** Also, the evaluator found that these sections include configuration and instructions of the protocol used to secure remote administrative session. Specifically, the AGD provides instructions for configuring the following protocols: SSH. Based on these findings, this assurance activity is considered satisfied.

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	• The audit records required for this test case can be found in the test cases associated with each of the listed SFRs.
Expected Test Results	• The TOE should be able to generate audit records for each of the events described in the ST under the FAU_GEN.1.2. • The audit records generated should match the proper format as specified in the guidance documentation.
Pass/Fail with Explanation	Pass. TOE is able to generate audit records for each of the management function specified in ST. This meets the testing requirements.

7.1.2 FAU_STG.1 Test #1

Item	Data
Test Assurance Activity	Test 1: The evaluator shall access the audit trail without authentication as Security Administrator (either by authentication as a non- administrative user, if supported, or without authentication at all) and attempt to modify and delete the audit records. The

	evaluator shall verify that these attempts 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 access the audit trail 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 into TOE as normal user. • Verify test is normal user. • Attempt to access/modify audit trail and verify command is rejected. • Verify via logs that normal user was able to login into TOE, view audit logs but not modify or delete them.
Expected Test Results	• TOE should not allow normal user to modify the audit records.
Pass/Fail with Explanation	Pass. Normal users are not able to modify the audit trail. This meets the testing requirements.

7.1.3 FAU_STG.1 Test #2

Item	Data
Test Assurance Activity	Test 2: The evaluator shall access the audit trail as an authorized administrator and attempt to delete the audit records. The evaluator shall verify that these attempts succeed. The evaluator shall verify that only the records authorized for deletion are deleted.
Test Steps	• Login into TOE as Security Administrator. • Verify diag is a Security Administrator. • Attempt to modify the audit records on the TOE it should be successful. • Verify via log that Security Administrator was able to login into TOE.
Expected Test Results	• TOE should allow Security Administrator to modify the audit records.
Pass/Fail with Explanation	Pass. The Security administrator is able to modify the audit trail. This meets the testing requirements.

7.1.4 FAU_STG_EXT.1 Test #1

Item	Data
Test Assurance Activity	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 Audit Server setting on TOE. • Verify audit server configuration is applied on the TOE. • Verify audit server version. • Configure rsyslog listener port on audit server. • Generate audit data on TOE. • Confirm that each event has been logged on to the audit server. • Verify by checking packet capture that TOE does not send data in Plaintext to the audit server.
Expected Test Results	• Evidence showing that logs generated on the TOE are the same as those transferred to the external audit server.
Pass/Fail with Explanation	Pass. The TOE is capable of transferring audit data to an external audit server automatically without administrator intervention. This meets the testing requirements.

7.1.5 FAU_STG_EXT.1 Test #2 (a)

Item	Data
Test Assurance Activity	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	NA. As per ST, TOE overwrite previous audit records when the local storage space for audit data is full.

7.1.6 FAU_STG_EXT.1 Test #2 (b)

Item	Data
Test Assurance Activity	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	• Observe the last archived log date and time is Mar 28 10:29 with the file name auth.log-20230328-1679999832. • Find the timestamp of the oldest message in the local audit log. • Generate lots of audit records with acumens script. • Verify that the audit logs have been generated and auth.log file filled up to 3MB. • Verify with the timestamp of the oldest message in the local audit log is overwritten with new one. • Observe the last archived log date and time is changed on Apr 14 08:05 with the file auth.log-20230414-1681459502.
Expected Test Results	• The TOE should overwrite the previous audit records when the local audit space is filled.
Pass/Fail with Explanation	Pass. The TOE is able to overwrite audit data once the audit storage is full to maximum. This meets the testing requirements.

7.1.7 FAU_STG_EXT.1 Test #2 (c)

Item	Data
Test Assurance Activity	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	NA. As per ST, TOE overwrite previous audit records when the local storage space for audit data is full. No other action.

7.1.8 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	• Check current time on TOE. • Configure remote NTP server entry on TOE. • Verify TOE is configured to sync with the remote NTP server. • Verify NTP server time updated on TOE. • Verify TOE and Remote NTP server connection is successful with NTP version 4 as seen in packet capture. • Verify with TOE log that TOE is able to sync with remote NTP server.
Expected Test Results	• TOE should be able to communicate with remote NTP server with correct version. • Evidence of configuration of TOE as NTP client. • Packet capture logs between TOE and remote NTP server showing correct version of NTP.
Pass/Fail with Explanation	Pass. The TOE uses the correct NTP version. This meets the testing requirement.

7.1.9 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	• Configure NTP authentication on TOE. • Configure proper authentication on the NTP server. • Verify that NTP authentication is successful.

	• Verify via Logs. • Verify with packet capture. • Check the available message digest algorithm used by NTP server. • Modify the message digest algorithm used by TOE. • Verify that NTP update fails. • Verify the failure with packet capture. • Verify via Logs.
Expected Test Results	• Connection should be failed as TOE is unable to synchronize time via NTP server.
Pass/Fail with Explanation	Pass. TOE does not sync time for NTP server if configured with unclaimed message digest algorithm. This meets the testing requirements.

7.1.10 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 current time on the TOE. • Check the current time on the NTP server. • Set NTP server to broadcast to 10.1.5.255. • Verify with packet capture that broadcast packets are sent by NTP server. • Verify that the time on the TOE is not modified. • Verify via logs. Multicast: • Check the current time on the TOE. • Check the current time on the NTP server. • Set NTP server to multicast to 224.0.1.1. • Verify with packet capture that multicast packets are sent by NTP server.

	• Check the time on TOE and verify that it is not modified due to NTP. • Verify via logs.
Expected Test Results	• Broadcast packets sent by the NTP server are not able to update the timestamp on the TOE as seen by comparing the NTP Broadcast Transmit time on the NTP server and the transmit time of the TOE in the packet capture. • Multicast packets sent by the NTP server are not able to update the timestamp on the TOE as seen by comparing the NTP Multicast Transmit time on the NTP server with the transmit time of the TOE in the packet capture.
Pass/Fail with Explanation	Pass. The TOE doesn't respond to NTP broadcast and multicast. This meets the testing requirement.

7.1.11 FCS_NTP_EXT.1.4 Test #1

Item	Data
Test Assurance Activity	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. **TD0528 applied**
Test Steps	• Verify the current time on TOE. • Configure 3 NTP server entry on TOE. • Verify TOE is configured to sync with 3 NTP servers. • Verify with packet capture. • Verify with TOE logs that TOE has successfully configured 3 NTP time source.
Expected Test Results	• Three NTP servers can be configured on the TOE as shown by the screenshots. • When three NTP servers are configured on the TOE, the TOE is able to successfully synchronize with all the time servers as is shown by packet captures which show NTP packets being received from each of the NTP servers and TOE logs for NTP version 4.
Pass/Fail with Explanation	Pass. The TOE is able to update the time using three NTP servers. This meets the testing requirement.

7.1.12 FCS_NTP_EXT.1.4 Test #2

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. **TD0528 applied**
Test Steps	• Verify the time on the TOE. • Configure an NTP server. • Sync the TOE with NTP server and capture those packets. • Verify with packet capture. • Verify with logs that TOE is sync to NTP server. • Configure a different NTP server to which the TOE syncs. • Verify that TOE is sync to different NTP server. • Verify with logs that TOE is sync to different NTP server. • Replay the packets from the NTP server which were captured during earlier sync. • Verify the TOE does not sync with the earlier NTP server.
Expected Output	• The timestamp on the TOE cannot be modified by an unconfigured or rogue NTP server. Packets captured and replayed later from a valid NTP server, which are the rogue packets do not have an effect on the TOE and the TOE does not respond to them or update the timestamp. This can be seen from the packet captures which show the replayed packets and the screenshots showing that the time on the TOE is not affected by the rogue packets.
Pass/Fail with Explanation	Pass. The TOE is accepting valid NTP server as time source while rejecting rouge NTP server as tine source. This meets the testing requirements.

7.1.13 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 shall use 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	• Login into TOE remotely via SSH and check current time. • Set new time on TOE via SSH. • Verify that time is updated on TOE. • Verify via TOE logs time is modified by Security administrator.
Expected Test Results	• The TOE should allow time to be set manually over SSH using the 'clock set' command. • Evidence (screenshot or CLI output) showing time configuration and time change. • Log showing each time change.
Pass/Fail with Explanation	Pass. The TOE allows the administrative user to configure the time on the TOE. This meets the testing requirements

7.1.14 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	• Check the current time on TOE. • Configure TOE as NTP client. • Check the updated time on TOE. • Verify with logs. • Verify via packet capture.
Expected Test Results	• The TOE is able to allow time updates from a configured NTP server. • Screenshots showing the time on the TOE updated. • Audit logs showing the time on the TOE is set by a NTP server.

Pass/Fail with Explanation	Pass. TOE support configuration of NTP server to be used as time source. This meets the testing requirements.

7.1.15 FPT_STM_EXT.1 Test #3

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	Pass. Not applicable, as the TOE does not obtain its time from the underlying VS.

7.1.16 FTP_ITC.1 Test #1

Item	Data
Test Assurance Activity	The evaluator 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.
Test Steps	This test is performed in conjunction with the test associated in FAU_STG_EXT.1 Test#1 and FTP_ITC.1 Test #4 for Audit server; and FPT_TUD_EXT.1 Test#1 and FTP_ITC.1 Test #4 for File server.
Pass/Fail with Explanation	Pass. External connections from the TOE are sent via an encrypted channel. This meets the testing requirements.

7.1.17 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.

Pass/Fail with Explanation	Pass. This test is performed in conjunction with the tests associated with FTP_ITC.1 Test# 1.

7.1.18 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.
Pass/Fail with Explanation	Pass. This test is performed in conjunction with the tests associated with FTP_ITC.1 Test# 1

7.1.19 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	Audit Server • Configure the TOE to establish secure channel with Audit Server. • Initiate the connection between the TOE and Audit Server. • Physically interrupt the connection between TOE and Audit Server for interval greater than application timeout and verify no data is sent in plain text for interval greater than application timeout using packet capture. • Verify via logs that physical connectivity was interrupted between the TOE and Audit Server.

	• Continue to attempt communication and re-connect the Audit Server and TOE after a short period of time. Verify communication return. • Physically interrupt the connection between TOE and Audit Server for interval less than application timeout and verify no data is sent in plain text for interval less than application timeout using packet capture. • Verify via logs the secure channel between TOE and Audit Server remain intact for interval less than application timeout. File Server • Configure the TOE to establish secure channel with File Server. • Initiate the connection between the TOE and File Server. • Physically interrupt the connection between TOE and File Server for interval greater than application timeout and verify no data is sent in plain text for interval greater than application timeout using packet capture. • Verify via logs that physical connectivity was interrupted between the TOE and File Server. • Continue to attempt communication and re-connect the File Server and TOE after a short period of time. Verify communication return. • Physically interrupt the connection between TOE and File Server for interval less than application timeout and verify no data is sent in plain text for interval less than application timeout using packet capture. • Verify via logs the secure channel between TOE and File Server remain intact for interval less than application timeout.
Expected Test Results	• The TOE should not send plaintext traffic when physical connectivity with a remote IT entity is interrupted and then restored.
Pass/Fail with Explanation	Pass. The TOE does not send plaintext traffic when disconnected from the Audit Server as well as File Server This meets the testing requirements.

7.2 Auth

7.2.1 FIA_AFL.1 Test #1

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. **TD0570 has been applied.**
Test Steps	• Configure Lock-out policy on TOE along with new user test. • Authenticate and verify incorrect credentials (test/asdfkjhqewrq) result in a failed connection. • Verify with logs that attempts with unsuccessful credentials will be rejected. • Login with good credentials (test/123TesT321) and verify that it fails. • Verify the same via logs.
Expected Test Results	• TOE rejects the connections once the maximum unsuccessful failure attempt is reached. • Evidence of configuration of lockout policy on TOE. • Log of user getting disconnected after maximum unsuccessful failure attempt.
Pass/Fail with Explanation	Pass. The TOE did not allow authentication once the authentication attempt limit has been reached. This meets the testing requirements.

7.2.2 FIA_AFL.1 Test #2a

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 administrator's access results in successful access (when using valid credentials for that administrator). **TD0570 has been applied.**
Test Steps	Time period selection is included in ST.

Pass/Fail with Explanation	NA. Administrator action is not selected in the ST.

7.2.3 FIA_AFL.1 Test #2b

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 authorization 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 authorization attempt using valid credentials results in successful access. **TD0570 has been applied.**
Test Steps	• Configure lock-out policy on TOE. • Try to login into TOE before lockout time with correct credentials it should fail. • Verify via logs that test user is not able to login due to lock-out policy. • Login into TOE after lockout time it should be success. • Verify via TOE logs that user is able to login successfully after lockout time.
Expected Test Results	• TOE logs verify that user is locked out for configured time after set successive unsuccessful authentication attempts. • TOE logs verify that user is able to login successfully after the lockout period.
Pass/Fail with Explanation	Pass. The TOE did not allow authentication until the configured lock-out time period had expired or an administrator unlocks the account. This meets the testing requirements.

7.2.4 FIA_PMG_EXT.1 Test #1

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. **TD0571 has been applied.**
Test Steps	• Create Password Policy on TOE. • Create User with Username: User1 and Password: QWERTYqwerty(1)2. • Verify User1 is created on TOE. • Verify with TOE logs user1 is created. • Create User with Username: User2 and Password: UIOPASuiopas!*34. • Verify User2 is created on TOE. • Verify with TOE logs user2 is created. • Create User with Username: User3 and Password: DFGHJKdfghjk@&56. • Verify User3 is created on TOE. • Verify with TOE logs user3 is created. • Create User with Username: User4 and Password: LZXClvxcm#^78. • Verify User4 is created on TOE. • Verify with TOE logs user4 is created. • Create User with Username: User5 and Password: VBNMbnz$%90. • Verify User5 is created on TOE. • Verify with TOE logs user5 is created. • Create User with Username: User6 and Password: A1+,-./:=;<_>\`~12z. • Verify User6 is created on TOE. • Verify with TOE logs user6 is created.
Expected Test Results	• TOE should allow evaluator to create users meeting password policy requirements configured on TOE. • Evidence of new user creation on TOE. • Logs of TOE to verify users created on TOE successfully.
Pass/Fail with Explanation	Pass. The TOE was able to create users with good passwords. The evaluator tested various combinations to cover uppercase, lowercase, numbers, and special characters. The evaluator was able to test minimum length and covered all special characters supported by the TOE. This meets the testing requirement.

7.2.5 FIA_PMG_EXT.1 Test #2

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. **TD0571 has been applied.**
Test Steps	• Create user bad1 with password length 9 having combination of special characters, uppercase characters, lowercase characters and numeric characters. This will fail. • Create user bad2 with password missing uppercase character. This will fail. • Create user bad3 with password missing lowercase character. This will fail. • Create user bad4 with password missing numeric character. This will fail. • Create user bad5 with password missing special character. This will fail.
Expected Test Results	• TOE should not allow evaluator to create users that is not matching the password policy configured on TOE. • Evidence of users are not created on TOE.
Pass/Fail with Explanation	Pass. The TOE rejected passwords that do not meet the password policy configured on TOE. This meets the testing requirements.

7.2.6 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	Console • Attempt to login from a local connection with incorrect credentials.

	• Verify that an audit records were generated showing login failure. • Log into the TOE from a local connection with correct credentials. • Verify that an audit records were generated showing login success. Remote-SSH Public key authentication • Test is covered by FCS_SSHS_EXT.1.2 Test#1 and FCS_SSHS_EXT.1.2 Test#2. Password-based authentication. • Attempt to login from a remote CLI connection with incorrect credentials. • Verify that an audit records were generated showing login failure. • Log into the TOE from a remote CLI connection with correct credentials. • Verify that an audit records were generated showing login success.
Expected Test Results	• The TOE should deny access when incorrect authentication presented and allow access when correct authentication credentials are presented. • Evidence (screenshot or CLI output) of each authentication attempt. • Logs showing successful/unsuccessful authentication attempts.
Pass/Fail with Explanation	Pass. Presenting incorrect authentication credentials results in denied access to the TOE. Presenting correct authentication credentials results in access being allowed to the TOE. This meets the testing requirements.

7.2.7 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-SSH Public key authentication

	• Connect to the TOE remotely over ssh using the incorrect public key and verify that only the TOE banner is displayed. The TOE falls back to password-based authentication and no other services are available. • Connect to the TOE remotely over ssh using the correct public key and verify that only the TOE banner is displayed and verify that the session is established. Password-based authentication. • Connect to the TOE remotely using ssh and verify the only option presented is the username/password entry and give incorrect credentials. • Attempt to connect to the TOE with correct credentials remotely and verify that the previously disabled commands are now available.
Expected Test Results	• No services except displaying a banner is available to a remote administrator attempting to login to the TOE via SSH.
Pass/Fail with Explanation	Pass. No system services are available to an unauthenticated user connecting remotely. This meets the testing requirements.

7.2.8 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	• Try to login into TOE locally with incorrect password and user is able to see only login banner. • Verify user failure with TOE logs. • Login into TOE locally with correct password and configuration functionality is provided on TOE. • Verify user login sucessfully with TOE logs.
Expected Test Results	• No services except displaying a banner is available to a local administrator attempting to login to the TOE. • Log showing inability to access any services prior to login.
Pass/Fail with Explanation	Pass. No system services are available to an unauthenticated user via the directly connected console. This meets the testing requirements.

7.2.9 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	• Connect to the TOE via console with correct authentication credentials and verify that the most obscured feedback is provided. • Verify authentication logs reflect success.
Expected Test Results	• The TOE should not provide anything other than obscured feedback at the directly connected login prompt. • Evidence (screenshot or CLI output) showing no output from the password being entered. • Logs show successful/unsuccessful login attempts.
Pass/Fail with Explanation	Pass. At the directly connected login prompt, the TOE does not provide anything more than no feedback. This meets the testing requirements.

7.2.10 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 in TOE via lower privileged user. • Attempt to access configuration mode without the proper privilege and verify user is unable to access it. • Attempt to perform an update command and verify the command is rejected. • Verify via TOE logs that normal user was able to login into TOE but not able to perform legitimate update.
Expected Test Results	• The TOE should reject attempts from an unprivileged user to update a legitimate image on the TOE. • Evidence (screenshot or CLI output) showing unsuccessful attempts.
Pass/Fail with Explanation	Pass. Unprivileged users cannot perform a software update on the TOE. This meets the testing requirements.

7.2.11 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.
Test Steps	This testing was performed in conjunction with in FPT_TUD_EXT.1 Test #1.
Pass/Fail with Explanation	Pass. This testing is covered by the requirements in FPT_TUD_EXT.1 Test #1.

7.2.12 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 in TOE as a lower privileged user. • Verify that "test" is a lower privileged user. • Attempt to modify the parameters involved with the syslog server and verify the command is rejected. • Verify via TOE logs that normal user was able to login into TOE.
Expected Test Results	• TOE should reject attempts from an unprivileged user to modify audit data on the TOE. • Evidence (CLI or Screenshot) of normal user not able to modify syslog parameters.
Pass/Fail with Explanation	Pass. Normal User is not able to modify syslog parameters after login on the TOE. This meets the testing requirements.

7.2.13 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 in TOE as Security Administrator. • Verify that "diag" is Security Administrator. • Try to modify syslog parameter on TOE it should be successful. • Verify via TOE logs that Security Administrator was able to login into TOE.
Expected Test Results	• The TOE should allow Security Administrator to modify syslog parameters. • Evidence (screenshot or CLI output) showing TOE is able modify syslog parameters. • Logs from TOE showing Security Administrator is able to modify the syslog parameters.
Pass/Fail with Explanation	Pass. The Security Administrator is able to modify the syslog parameters on the TOE. This meets the testing requirements.

7.2.16 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 in TOE as a lower privileged user. • Verify that "test" is a lower privileged user.

	• Attempt to install SSH session keys on TOE and verify the command is rejected. • Verify via TOE logs that normal user was able to login into TOE.
Expected Test Results	• The TOE should reject attempts from an unprivileged user to modify, delete, generate/import crypto keys on the TOE. • Evidence (screenshot or CLI output) showing privilege level of the user. • Evidence (screenshot or CLI output) showing unsuccessful attempts.
Pass/Fail with Explanation	Pass. Unprivileged users cannot modify the cryptographic keys on the TOE. This meets the testing requirements.

7.2.17 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 in TOE as Security Administrator. • Verify that “diag” is Security Administrator. • Attempt to install SSH session keys on TOE. Verify it succeeds. • Verify via TOE logs that Security Administrator was able to login into TOE.
Expected Test Results	• The TOE should accept attempts from a privileged user to modify, delete, generate/import crypto keys on the TOE. • Evidence (screenshot or CLI output) showing privilege level of the user. • Evidence (screenshot or CLI output) and log showing successful attempts.
Pass/Fail with Explanation	Pass. Security Administrator is able to modify the cryptographic keys on TOE. This meets the testing requirements.

7.2.18 FMT_SMF.1 Test #1

Item	Data
Test Assurance Activity	The evaluator shall test 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 Steps	The TSF shall be capable of performing the following management functions:

	• *Ability to administer the TOE locally and remotely;*
	• *Ability to configure the access banner;*
	• *Ability to configure the session inactivity time before session termination or locking;*
	• *Ability to update the TOE, and to verify the updates using [signature] capability prior to installing those updates;*
	• *Ability to configure the authentication failure parameters for FIA_AFL.1;*
	• [
	  ○ *Ability to modify the behaviour of the transmission of audit data to an external IT entity;*
	  ○ *Ability to manage the cryptographic keys;*
	  ○ *Ability to configure thresholds for SSH rekeying;*
	  ○ *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 key database;*
	  ○ *No other capabilities*].

Expected Test Results	• All management functions identified in Security Target should be met by presenting correct test cases.
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. Therefore, this test is passed.

7.2.19 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.
Test Steps	There are two interfaces where these can be tested (over the CLI and remote SSH). It is covered in FIA_UIA_EXT.1.1 Test #2, FIA_UIA_EXT.1.1 Test #3, FTA_SSL_EXT.1.1 Test #1, FTA_SSL.3.1 Test #1, FTA_SSL.4.1 Test #1 and FTA_TAB.1 Test #1. This meets the testing requirements.

Pass/Fail with Explanation	Pass. There are two interfaces where these can be tested (over the CLI and remote SSH). It is covered in FIA_UIA_EXT.1.1 Test #2, FIA_UIA_EXT.1.1 Test #3, FTA_SSL_EXT.1.1 Test #1, FTA_SSL.3.1 Test #1, FTA_SSL.4.1 Test #1 and FTA_TAB.1 Test #1. This meets the testing requirements.

7.2.20 FTA_SSL.3 Test #1

Item	Data
Test Assurance Activity	The evaluator shall follow 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	• Configure idle timeout of 60 seconds. • Login into TOE remotely via SSH. • Check the Time on TOE. • Verify with TOE logs user login success and time. • Keep the SSH session idle for 60 seconds and verify with TOE logs that session terminated due session idle timeout. • Configure idle timeout of 120 seconds. • Login into TOE remotely via SSH. • Check the Time on TOE. • Verify with TOE logs user login success and time. • Keep the SSH session idle for 120 seconds and verify with TOE logs that session terminated due session idle timeout.
Expected Test Results	• The TOE should terminate idle remote sessions after the configured time. • Evidence (e.g., screenshot or CLI output) showing configuration of time out value. • Log showing the administrative log on (with time). • Evidence (e.g., screenshot or CLI output) showing administrator being terminated. • Log showing the termination of the connection.
Pass/Fail with Explanation	Pass. The remote administrative time out periods can be set by the administrative user. The TOE enforces the configured inactivity period in each instance. This meets the testing requirements.

7.2.21 FTA_SSL.4 Test #1

Item	Data
Test Assurance Activity	The evaluator shall initiate 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	• Login into TOE locally. • Verify the logs reflect log in. • Using the instructions provided by the user guide log off. • Verify the logs reflect the log off.
Expected Test Results	• The TOE should allow the user to terminate the directly connected administrative sessions. • Evidence (e.g., screenshot or CLI output) showing logging into the TOE locally. • Evidence (e.g., screenshot or CLI output) showing the log out. • Log showing the log out.
Pass/Fail with Explanation	Pass. The TOE allows user to terminate the directly connected administrative sessions. This meets the testing requirements.

7.2.22 FTA_SSL.4 Test #2

Item	Data
Test Assurance Activity	The evaluator shall initiate 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	• Login into TOE remotely via SSH. • Verify with TOE logs remote login via SSH is successful. • Terminate the remote session of TOE manually. • Verify with TOE logs remote session terminated by user.
Expected Test Results	• The TOE should allow the user to terminate the interactive remote sessions. • Evidence (e.g., screenshot or CLI output) showing logging into the TOE remotely. • Evidence (e.g., screenshot or CLI output) showing the log out. • Log showing the log out.

Pass/Fail with Explanation	Pass. The TOE allows user to terminate the remote administrative sessions. This meets the testing requirements.

7.2.23 FTA_SSL_EXT.1.1 Test #1

Item	Data
Test Assurance Activity	The evaluator shall follow 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	• Login into TOE locally. • Configure idle timeout of 60 seconds. • Check the Time on TOE. • Wait for 62 seconds and run a command. • Verify via TOE logs session terminated due to inactivity time period. • Configure a new idle timeout of 120 seconds. • Check the Time on TOE. • Wait for 122 seconds and run a command. • Verify via TOE logs session terminated due to inactivity time period.
Expected Test Results	• The TOE should terminate idle local sessions after the configured time. • Evidence (e.g., screenshot or CLI output) showing configuration of time out value. • Log showing the administrative log on (with time). • Evidence (e.g., screenshot or CLI output) showing administrator being terminated. • Log showing the termination of the connection.
Pass/Fail with Explanation	Pass. The local administrative inactivity was able to be set to multiple values. In each instance, the TOE logged the user out after the configured time. This meets the testing requirements.

7.2.24 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 access banner on TOE. • Login into TOE via SSH it should display access banner. • Verify with TOE logs that the access banner is configured on TOE. • Verify with TOE logs that the user successfully logged in via SSH. • Login into TOE via console and verify banner is visible while login.
Expected Test Results	• When any user accesses the TOE through the console and SSH, the configured banner should be displayed prior to authenticating the TOE. • Evidence (e.g., screenshot or CLI output) showing configuration of access banners. • Log showing configuration of the access banners. • Evidence (e.g., screenshot or CLI output) from logon with access banners.
Pass/Fail with Explanation	Pass. An access banner can be set for all the methods that can be used to access the device. This meets the testing requirements.

7.2.25 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	• Log into TOE remotely via SSH. • Verify that Wireshark shows a successful connection and data is not sent in plaintext. • Verify that the TOE shows a successful connection.

Expected Test Results	• The TOE should establish communication between TOE and remote administrator. • Evidence (screenshot or CLI output) showing attempt to connect via the trusted paths. • Log showing successful connection.
Pass/Fail with Explanation	Pass. Remote administrative access to the TOE is over secure protected channels and the data was not sent in plaintext. This meets the testing requirements.

7.2.26 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.
Test Steps	This is covered by FTP_TRP.1/Admin_T1 and FCS_SSH_EXT.1. In that test, the data was not sent in plaintext.
Pass/Fail with Explanation	Pass. This is covered by FTP_TRP.1/Admin_T1 and FCS_SSH_EXT.1. In that test, the data was not sent in plaintext.

7.3 SSH

7.3.1 FCS_SSHS_EXT.1.2 Test #1

Item	Data
Test Assurance Activity	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. **TD0631 applied.**
Test Steps	• Generate an ssh-rsa public key on the VM. • Copy the public key onto the TOE and verify that it is updated on the TOE. • Login to the TOE using the public key and verify that the session is established. • Verify via logs that the session was established using the configured public key. • Verify via packet capture.

	• Generate an ecdsa-sha2-nistp256 public key on the VM. • Copy the public key onto the TOE and verify that it is updated on the TOE. • Login to the TOE using the public key and verify that the session is established. • Verify via logs that the session was established using the configured public key. • Verify via packet capture.
Expected Test Results	• TOE logs show successful establishment of the SSH connection. • Evidence (screenshot or CLI output) showing successful SSH connection. • TOE logs show successful authentication of user.
Pass/Fail with Explanation	Pass. The remote client is able to establish a successful SSH connection using each one of the supported public key algorithms. This meets the testing requirements.

7.3.2 FCS_SSHS_EXT.1.2 Test #2

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. **TD0631 applied.**
Test Steps	• Generate a new client key pair with ecdsa-sha2-nistp256 on the VM. • Login to the device using public key without updating the public key on the TOE and verify that the connection fails. • Verify via audit logs that the connection fails. • Verify via packet capture that the connection fails.
Expected Test Results	• Verify that TOE denies authentication attempt from a client whose public key does not match the public key associated with it on the TOE. • Evidence (screenshot or CLI output) showing unsuccessful SSH connection. • TOE logs show unsuccessful authentication of user.
Pass/Fail with Explanation	Pass. The TOE is not able to establish a connection with a remote SSH client when the TOE is not configured to recognize the associated public key for authentication. This meets the testing requirements.

7.3.3 FCS_SSHS_EXT.1.2 Test #3

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. **TD0631 applied.**
Test Steps	• Configure the TOE to ensure that the TOE supports password-based authentication. • Log into the TOE via SSH with password authentication. • Verify the Audit logs. • Verify via TOE logs that that SSH session was established.
Expected Test Results	• User authentication to TOE using correct password is successful. • Evidence (screenshot or CLI output) of configuring password-based authentication. • Packet capture of session being established. • Log showing successful authentication.
Pass/Fail with Explanation	Pass. The TOE is able to establish a connection with a remote SSH user when correct authentication credentials are presented. This meets the testing requirements.

7.3.4 FCS_SSHS_EXT.1.2 Test #4

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. **TD0631 applied.**
Test Steps	• Configure the TOE to ensure that the TOE supports password-based authentication. • Attempt to Log into the TOE via SSH with password-based authentication parameters and provide incorrect password (This will fail).

	• Verify authentication logs reflect failures. • Verify via Packet Capture.
Expected Test Results	• User authentication to TOE using incorrect password results in failure. • TOE logs show unsuccessful authentication attempt by user.
Pass/Fail with Explanation	Pass. The TOE is not able to establish a connection with a remote SSH client when the TOE is not configured to recognize the associated public key for authentication. 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	• Use the acumen-sshs to start an SSH session with the TOE and send bad length packet. • Verify via packet capture that connection between TOE and remote SSH client is dropped. • Verify with TOE logs that TOE drop the large packet.
Expected Test Results	• Evidence (screenshot or CLI output) showing packets is discarded because window size is full. • TOE logs verify that packet larger than 256 KB is discarded with a 'Bad packet length' error.
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	• Verify that TOE supports AES128-ctr for encryption algorithm. • Establish an SSH session with the configured supported algorithms. • Verify successful establishment of connection via audit log. • Verify AES128-ctr was used via packet capture. • Verify that the TOE only supports all algorithms as mentioned in the ST via packet capture. • Verify that TOE supports AES256-ctr for encryption algorithm. • Establish an SSH session with the configured supported algorithms. • Verify successful establishment of connection via audit log. • Verify AES256-ctr was used via packet capture. • Verify that TOE supports AES128-gcm@openssh.com for encryption algorithm. • Establish an SSH session with the configured supported algorithms. • Verify successful establishment of connection via audit log. • Verify AES128-gcm@openssh.com was used via packet capture. • Verify that TOE supports AES256-gcm@openssh.com for encryption algorithm. • Establish an SSH session with the configured supported algorithms. • Verify successful establishment of connection via audit log. • Verify AES256-gcm@openssh.com was used via packet capture.
Expected Test Results	• Packet Capture shows TOE establishing successful connection with supported cipher. • Evidence (screenshot or CLI output) showing configuration of each algorithm. • Log showing successful/unsuccessful connection of each algorithm. • Packet capture showing successful/unsuccessful connection of each algorithm.
Pass/Fail with Explanation	Pass. TOE is able to establish connection from remote SSH server using claimed chippers. This meets the testing requirements.

7.3.7 FCS_SSHS_EXT.1.5 Test #1

Item	Data
Test Assurance Activity	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. **TD0631 applied.**
Test Steps	• Verify that the claimed host public key algorithms are enabled on TOE. • Establish a session with the TOE using the ssh-rsa host key algorithm. • Verify via logs that the session was established. • Verify via packet capture that the configured host key algorithm was used. • Establish a session with the TOE using ecdsa-sha2-nistp256 host key algorithm. • Verify via logs that the session was established. • Verify via packet capture that the configured host key algorithm was used.
Expected Test Results	• TOE logs show successful establishment of the SSH connection. • Packet capture shows session establishment with the configured host key algorithm. • Log showing successful connection of each algorithm.
Pass/Fail with Explanation	Pass. The remote client is able to establish 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

Item	Data
Test Assurance Activity	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. **TD0631 applied.**
Test Steps	• Verify that the claimed host public key algorithms are enabled on TOE. • Established a session with the TOE using the non-supported host key algorithms (SSH-ED25519). • Verify that the connection is refused via packet capture.

	• Verify that the SSH session was refused using ssh-ed25519 via log.
Expected Test Results	• TOE logs verify connection establishment using unsupported public key algorithm(ssh-ed25519) is denied by TOE. • Packet Capture verifies connection establishment using unsupported public key algorithm(ssh-ed-25519) is denied by TOE.
Pass/Fail with Explanation	Pass. TOE is able to reject the connection from remote client due to unsupported host public key algorithm. This meets the testing requirements.

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	• Verify that TOE supports HMAC-SHA1 for hashing algorithm. • Establish an SSH session with the configured supported algorithms (HMAC-SHA1). • Verify that the SSH session was encrypted using HMAC-SHA1 via capture. • Verify successful establishment of connection via log. • Verify that TOE supports HMAC-SHA2-256 for hashing algorithm. • Establish an SSH session with the configured supported algorithms (HMAC-SHA2-256). • Verify that the SSH session was encrypted using HMAC-SHA2-256 via capture. • Verify successful establishment of connection via log. • Verify that TOE supports HMAC-SHA2-512 for hashing algorithm. • Establish an SSH session with the configured supported algorithms (HMAC-SHA2-512). • Verify that the SSH session was encrypted using HMAC-SHA2-512 via capture. • Verify successful establishment of connection via log.

Expected Test Results	• Evidence (screenshot or CLI output) showing configuration of each algorithm. • Log showing successful/unsuccessful connection of each algorithm. • Packet capture showing successful/unsuccessful connection of each algorithm.
Pass/Fail with Explanation	Pass. The TOE is able to make 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 mac. • Verify via logs that the session fails due to unsupported mac algorithm. • Verify via packet capture that the TOE does not continue negotiation.
Expected Test Results	• TOE logs show unsuccessful negotiation with unsupported MAC algorithm(hmac-md5). • Packet Capture shows unsuccessful negotiation with unsupported MAC algorithm(hmac-md5).
Pass/Fail with Explanation	Pass. TOE does not support unauthorized hashing algorithm to establish SSH session from remote ssh client to itself. 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 an SSH session using diffiehellman-group1-sha1.

	• Verify that the SSH session was refused via logs. • Verify the connection is refused via packet capture.
Expected Test Results	• TOE logs show unsuccessful negotiation with diffiehellman-group1-sha1 key exchange. • Packet Capture shows unsuccessful negotiation with diffiehellman-group1-sha1 key exchange.
Pass/Fail with Explanation	Pass. The TOE rejects SSH connections using diffiehellman-group1-sha1 (a non-approved algorithm) for key exchange. 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	• Establish an SSH session with the configured supported key exchange algorithm (Diffie-hellman-group14-sha1). • Verify that the session is established via logs. • Verify that the SSH session was encrypted using Diffie-hellman-group14-sha1 via capture. • Establish an SSH session with the configured supported key exchange algorithm (Diffie-hellman-group14-sha256). • Verify that the session is established via logs. • Verify that the SSH session was encrypted using Diffie-hellman-group14-sha256 via capture. • Establish an SSH session with the configured supported key exchange algorithm (Diffie-hellman-group16-sha512). • Verify that the session is established via logs. • Verify that the SSH session was encrypted using Diffie-hellman-group16-sha512 via capture. • Establish an SSH session with the configured supported key exchange algorithm (ecdh-sha2-nistp256). • Verify that the session is established via logs. • Verify that the SSH session was encrypted using ecdh-sha2-nistp256 via capture.

	• Establish an SSH session with the configured supported key exchange algorithm (ecdh-sha2-nistp384). • Verify that the session is established via logs. • Verify that the SSH session was encrypted using ecdh-sha2-nistp384 via capture. • Establish an SSH session with the configured supported key exchange algorithm (ecdh-sha2-nistp521). • Verify that the session is established via logs. • Verify that the SSH session was encrypted using ecdh-sha2-nistp521 via capture
Expected Test Results	• TOE logs show successful negotiation with supported key exchange algorithm. • Packet Capture shows successful negotiation with supported key exchange algorithm.
Pass/Fail with Explanation	Pass. The TOE is able to make SSH connections with each claimed data key exchange method. This meets the testing requirements.

7.3.13 FCS_SSHS_EXT.1.8 Test #1t

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	• Login to the TOE remotely via SSH as a lower privileged user. • Verify that “test” is a lower privileged user. • Attempt to configure rekey parameters on TOE and verify command is rejected.

	• Verify via TOE logs that a lower privileged user was able to login into TOE. • Login to the TOE remotely via SSH as Security administrator. • Verify that "diag" is Security administrator. • Configure time based rekey of 600 seconds. Verify that command is accepted. • Verify via TOE logs that Security administrator is able to configure rekey. • Send a continuous ping and verify that a rekey generates every 600 Seconds. • Verify the login time for rekey. • Verify rekey via audit logs.
Expected Test Results	• TOE log verifies rekeying is initiated after 600 seconds. • Evidence (screenshot or CLI output) showing configuration of rekey for time. • Log showing session rekey request being sent after time-based threshold has been reached.
Pass/Fail with Explanation	Pass. The TOE initiates a rekey every 10 Minutes. 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: 1. An argument is present in the TSS section describing this hardware- based limitation and 2. 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	• Login to the TOE remotely via SSH as a lower privileged user. • Verify that "test" is a lower privileged user. • Attempt to configure rekey parameters on TOE and verify command is rejected. • Verify via TOE logs that a lower privileged user was able to login into TOE. • Login to the TOE remotely via SSH as Security administrator. • Verify that "diag" is Security administrator. • Configure traffic based rekey of 1G. Verify that command is accepted. • Verify via TOE logs that Security administrator is able to configure rekey. • Copy the file from Source to other TOE which is above 1GB in size to occur rekey. • Verify the login time for rekey. • Verify that a rekey is generated in every 1GB of data.
Expected Test Results	• TOE log verifies rekeying is initiated after 1GB of data transfer. • Evidence (screenshot or CLI output) showing configuration of rekey for volume threshold. • Log showing session rekey request being sent after volume-based threshold has been reached.
Pass/Fail with Explanation	Pass. The TOE initiates a rekey in every 1GB of data. This meets the testing requirements.

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	• Establish a connection with the TOE over TLS using the cipher RSA-AES_128-CBC-SHA and show the connection being successful. • Verify using pcap the required cipher suite TLS_RSA_WITH_AES_128_CBC_SHA. • Verify with logs. • Establish a connection with the TOE over TLS using the cipher RSA-AES_256-CBC-SHA and show the connection being successful. • Verify using pcap the required cipher suite TLS_RSA_WITH_AES_256_CBC_SHA. • Verify with logs. • Establish a connection with the TOE over TLS using the cipher ECDHE-RSA-AES128-CBC-SHA and show the connection being successful. • Verify using pcap the required cipher suite ECDHE-RSA-AES128-CBC-SHA. • Verify with logs. • Establish a connection with the TOE over TLS using the cipher ECDHE-RSA-AES256-CBC-SHA and show the connection being successful. • Verify using pcap the required cipher suite ECDHE-RSA-AES256-CBC-SHA. • Verify with logs. • Establish a connection with the TOE over TLS using the cipher RSA-AES128-CBC-SHA256 and show the connection being successful. • Verify using pcap the required cipher suite RSA-AES128-CBC-SHA256. • Verify with logs. • Establish a connection with the TOE over TLS using the cipher RSA-AES256-CBC-SHA256 and show the connection being successful. • Verify using pcap the required cipher suite RSA-AES256-CBC-SHA256. • Verify with logs. • Establish a connection with the TOE over TLS using the cipher RSA-AES128-GCM-SHA256 and show the connection being successful. • Verify using pcap the required cipher suite RSA-AES128-GCM-SHA256.

	• Verify with logs. • Establish a connection with the TOE over TLS using the cipher RSA-AES256-GCM-SHA384 and show the connection being successful. • Verify using pcap the required cipher suite RSA-AES256-GCM-SHA384. • Verify with logs. • Establish a connection with the TOE over TLS using the cipher ECDHE-RSA-AES128-GCM-SHA256 and show the connection being successful. • Verify using pcap the required cipher suite ECDHE-RSA-AES128-GCM-SHA256. • Verify with logs. • Establish a connection with the TOE over TLS using the cipher ECDHE-RSA-AES256-GCM-SHA384 and show the connection being successful. • Verify using pcap the required cipher suite ECDHE-RSA-AES256-GCM-SHA384. • Verify with logs. • Establish a connection with the TOE over TLS using the cipher ECDHE-RSA-AES128-CBC-SHA256 and show the connection being successful. • Verify using pcap the required cipher suite ECDHE-RSA-AES128-CBC-SHA256. • Verify with logs. • Establish a connection with the TOE over TLS using the cipher ECDHE-RSA-AES256-CBC-SHA384 and show the connection being successful. • Verify using pcap the required cipher suite ECDHE-RSA-AES256-CBC-SHA384. • Verify with logs.
Expected Test Results	• TOE should establish a connection with claimed ciphersuite.
Pass/Fail with Explanation	Pass. The TOE was able to make connections using the supported cipher suite. This meets the testing 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	• Create a server certificate with the Server Authentication EKU. • Attempt a connection from the TOE to a TLS server using the certificate that contains the Server Authentication EKU. • Verify with logs. • Verify that the TOE accepts the connection with packet capture. • Create a server certificate that lacks the Server Authentication EKU. • Attempt a connection from the TOE to a TLS server which is using the invalid certificate missing the Server Authentication EKU. • Verify with logs. • Verify that the TOE rejects the connection with packet capture.
Expected Test Results	• TOE should establish a connection with a server with authorized server certificate otherwise TOE should reject the connection. • Evidence (screenshot or CLI output) showing successful/unsuccessful TLS connection. • Packet capture shows successful/unsuccessful connection.
Pass/Fail with Explanation	Pass. The TOE accepted a connection with valid server certificate and denied a connection to a server which is using an invalid certificate. This meets the testing 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	• Attempt a connection with a server with a certificate that does not match the server-selected ciphersuite using acumen-tlsc tool. Verify that it fails.
	• Verify the connection fails with packet capture.
	  ○ TOE accepts RSA ciphers.
	  ○ Server sends EC cert.
	• Verify with logs.

Expected Test Results	• The TOE should be unable to establish a connection with non-supported ciphersuite. • Evidence (screenshot or CLI output) showing unsuccessful connection with unsupported ciphersuite. • Packet capture shows unsuccessful connection.
Pass/Fail with Explanation	Pass. The TOE denied a connection to a server using a certificate that doesn't match the cipher suite. 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	• Attempt a connection to a server using the TLS_NULL_WITH_NULL_NULL ciphersuite using acumen-tlsc tool. • Verify the connection is refused via packet capture. • Verify the Reason for failure via logs.
Expected Test Results	• The TOE should reject a connection when server selects non-supported algorithm. • Show TOE denies the connection with Packet capture. • Evidence (screenshot or CLI output) showing unsuccessful TLS connection.
Pass/Fail with Explanation	Pass. The TOE denied the connection to a server using a TLS_NULL_WITH_NULL_NULL ciphersuite. This meets the testing requirement.

7.4.5 FCS_TLSC_EXT.1.1 Test #4b

Item	Data
Test Assurance Activity	The evaluator shall 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	• Attempt a connection from the TOE to a remote TLS server using acumen-tlsc tool that would allow the server's ciphersuite to be modified. Verify that the connection fails. • Verify via packet Capture that the client rejects the connection after receiving the Server hello. • Verify with logs.
Expected Test Results	• Evidence (screenshot or CLI output) showing client rejecting a connection when server modifies a ciphersuite.

	• Packet capture shows unsuccessful connection.
Pass/Fail with Explanation	Pass. The modified TLS connection was rejected. This meets the testing requirement.

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	• Attempt a connection from the TOE to a remote TLS server using acumen-tlsc tool that will send an ECDHE key exchange using an unsupported curve. Verify that the connection fails. • Verify that the TOE disconnects after receiving the server's key exchange handshake message. • Verify with logs.
Expected Test Results	• TOE should reject the connection if an unsupported algorithm is provided. • Evidence (screenshot or CLI output) shows TOE rejecting connection if unsupported algorithm is provided. • TOE logs show unsuccessful connection.
Pass/Fail with Explanation	Pass. The TOE rejected a connection when an unsupported curve was used. This meets testing requirements.

7.4.7 FCS_TLSC_EXT.1.1 Test #5a

Item	Data
Test Assurance Activity	The evaluator shall 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	• Using acumen-tlsc tool, attempt a connection to a remote TLS server using a non-supported TLS version and verify that the TOE rejects the connection. • Verify the connection fails with packet capture. • Verify with logs.
Expected Test Results	• TOE should reject the connection when server sends a message with non-supported TLS version. • Evidence (screenshot or CLI output) showing TOE is rejecting a connection with unsupported TLS version. • TOE logs show unsuccessful connection because of unsupported TLS version.

Pass/Fail with Explanation	Pass. The TOE rejects a connection with a server using a non-supported TLS version. This meets the testing requirement.

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	• Attempt a connection from the TOE to a remote TLS server using acumen-tlsc tool that would allow the server's signature block to be modified. Verify that the connection fails. • Verify the connection with packet capture. • Verify the connection fails with logs.
Expected Test Results	• The connection establishment should fail when server's key exchange handshake message is modified. • Evidence (screenshot or CLI output) showing TOE rejecting connection when handshake message is modified. • TOE logs show unsuccessful connection.
Pass/Fail with Explanation	Pass. The TOE rejects the connection when the signature block is modified. This meets testing requirements.

7.4.9 FCS_TLSC_EXT.1.1 Test #6a

Item	Data
Test Assurance Activity	The evaluator shall 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	• Using acumen-tlsc tool modify the server finished handshake message. • Verify with packet capture. • Verify with logs.
Expected Test Results	• TOE should reject a connection when tool modifies server finished handshake message. • Evidence (screenshot or CLI output) showing TOE is rejecting a connection when message is modified. • TOE logs show unsuccessful connection because digest check failed.

Pass/Fail with Explanation	Pass. The modified TLS connection was rejected. This meets the testing requirements.

7.4.10 FCS_TLSC_EXT.1.1 Test #6b

Item	Data
Test Assurance Activity	The evaluator shall 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	• Attempt a connection to a modified TLS server that would allow sending a garbled message from the server after the server issues the ChangeCipherSpec message and verify that the TOE rejects the connection. • Verify with packet capture. • Verify with logs.
Expected Test Results	• Handshake should not happen when TOE receives garbled message.
Pass/Fail with Explanation	Pass. The TOE closes the connection after receiving garbled data. This meets the test requirements.

7.4.11 FCS_TLSC_EXT.1.1 Test #6c

Item	Data
Test Assurance Activity	The evaluator shall 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	• Using acumen-tlsc tool send modified nonce in the Server Hello handshake message to TOE. • Verify that the client rejects the Server Key Exchange handshake message using packet capture. • Verify via logs.
Expected Test Results	• Client should reject the handshake message when nonce in the server hello handshake is changed.
Pass/Fail with Explanation	Pass. The modified TLS connection was rejected. This meets the testing requirement.

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	Note- IP is not supported in CN. CN: FQDN • Configure the correct reference identifier in the TOE. • Create a server certificate with invalid CN and no SAN. • Connect to the TLS Server using the mismatched CN and verify that it fails. • Verify with packet capture. • Verify with logs.
Expected Test Results	• TOE should reject a connection when reference identifier does not match with server certificate CN.
Pass/Fail with Explanation	Pass. The TOE did not establish a connection when presented with a server certificate that does not contain an identifier in the Common Name (CN) that matches the reference identifier and does not contain a Subject Alternative Name (SAN). 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	Note: As per test objective evaluator need to perform test for each supported SAN ID types. SAN: IPv4 address • Configure the correct reference identifier in the TOE. • Create a server certificate with valid CN but invalid SAN. • Attempt a connection to the TLS server and verify that it fails. • Verify with packet capture. • Verify with logs. SAN: FQDN • Configure the correct reference identifier in the TOE. • Create a server certificate with valid CN but invalid SAN. • Attempt a connection to the TLS server and verify that it fails. • Verify with packet capture. • Verify with logs.
Expected Test Results	• TOE should reject a connection when reference identifier does not match with server certificate SAN.
Pass/Fail with Explanation	Pass. The TOE rejects certificates with a good CN but bad SAN. This meets the testing requirement.

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	Note- IP is not supported in CN. CN: FQDN • Configure the correct reference identifier in the TOE. • Create a server certificate with correct CN and no SAN. • Connect to the TLS Server and verify that the connection is established. • Verify with packet capture that connection is successful. • Verify via logs.
Expected Test Results	• TOE should accept a connection when matching CN is identified even though SAN field is missing from the certificate.
Pass/Fail with Explanation	Pass. The TOE establish a connection when presented with a server certificate that does contain an identifier in the Common Name (CN) that matches the reference identifier and does not contain a Subject Alternative Name (SAN). 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	SAN: IPv4 address • Configure the correct reference identifier in the TOE. • Create a server certificate with incorrect CN and valid SAN. • Connect to the TLS Server and verify that the connection is established. • Verify with packet capture that connection is successful. • Verify via logs. SAN: FQDN • Configure the correct reference identifier in the TOE. • Create a server certificate with incorrect CN and valid SAN. • Connect to the TLS Server and verify that the connection is established. • Verify with packet capture that connection is successful.

	• Verify via logs.
Expected Test Results	• TOE should accept a connection when matching SAN is identified even though CN reference identifier does not match
Pass/Fail with Explanation	Pass. A connection was established when TOE is presented with a server certificate that contains a CN that does not match the reference identifier but does contain an identifier in the SAN that matches. 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 correct reference identifier in the TOE. • Create a server certificate containing a wildcard that is not in the left-most label of CN. • Verify that the connection fails. • Verify with packet capture. • Verify with logs. SAN • Configure the correct reference identifier in the TOE. • Create a server certificate containing a wildcard that is not in the left-most label of SAN. • Verify that the connection fails. • Verify with logs. • Verify with packet capture.
Expected Test Results	• TOE should reject a connection when wildcard is not in the left-most position in SAN.
Pass/Fail with Explanation	Pass. The connection fails when TOE is presented with a server certificate containing a wildcard that is not in the left most label of the CN or SAN. 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 correct reference identifier on the TOE. • Create a server certificate without the left-most label of the CN. • Attempt to connect to the TOE and verify that the connection is successful. • Verify with packet capture. • Verify with logs. SAN • Configure the correct reference identifier on the TOE. • Create a server certificate without left-most label of the SAN. • Attempt to connect to the TOE and verify that the connection is successful. • Verify with packet capture. • Verify with logs.
Expected Test Results	• TOE should accept the connection when wildcards are supported.

Pass/Fail with Explanation	Pass. The TOE established a connection with a server having a wildcard configured in the single leftmost label of the CN or the SAN. This meets the testing requirement.

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 correct reference identifier on the TOE. • Create a server certificate with a wildcard in the leftmost label of CN. • Attempt to connect to the TOE and verify that the connection fails. • Verify with packet capture. • Verify with logs. SAN: • Configure the correct reference identifier on the TOE. • Create a server certificate with a wildcard in the leftmost label of SAN. • Attempt to connect to the TOE and verify that the connection fails. • Verify with packet capture. • Verify with logs.
Expected Test Results	• TOE should reject a connection when TOE does not support wildcards.

Pass/Fail with Explanation	Pass. The TOE rejects a connection with a server when the reference identifier is without the left most label in the CN and SAN. 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 correct reference identifier on the TOE. • Create a server certificate with a wildcard in the leftmost label of CN. • Attempt to connect to the TOE and verify that the connection fails. • Verify with packet capture. • Verify with logs. SAN: • Configure the correct reference identifier on the TOE. • Create a server certificate with a wildcard in the leftmost label of SAN. • Attempt to connect to the TOE and verify that the connection fails. • Verify with packet capture. • Verify with logs.
Expected Test Results	• The connection should be failed when reference identifier is configured with two left-most labels.

Pass/Fail with Explanation	Pass. The TOE rejects a connection when the server certificate contains a wildcard in the CN, when the CN is an IP address. This meets the testing requirement.

7.4.20 FCS_TLSC_EXT.1.2 Test #6

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 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 Test6. **TD0790 applied**
Test Steps	• Configure the correct reference identifier on the TOE. • Create a server certificate with a CN that matches the reference identifier but replace one of the groups with an *. • Attempt a connection with the TOE and verify that it fails. • Verify with packet capture. • Verify with logs that connection fails.
Expected Test Results	• Connection should be failed when the server certificate contains a wildcard in the CN, when the CN is an IP address but the SAN is not present..
Pass/Fail with Explanation	Pass. The TOE rejects a connection when the server certificate contains a wildcard in the CN, when the CN is an IP address but the SAN is not present. This meets the testing requirement.

7.4.21 FCS_TLSC_EXT.1.2 Test #7

Item	Data
Test Assurance Activity	The evaluator shall configure the reference identifier according to the AGD guidance and perform the following tests during a TLS connection: Test 7 [conditional]: 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): 1) 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. 2) 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. 3) 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, the TOE is a standalone TOE and does not claim or use secure channel for FPT_ITT.

7.4.22 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.

Pass/Fail with Explanation	Pass. Test covered by FIA_X509_EXT.1.1/Rev Test #1.

7.4.23 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.
Pass/Fail with Explanation	Pass. Test covered by FIA_X509_EXT.1.1/Rev Test #1(TLS), FIA_X509_EXT.1.1/Rev Test #2(TLS), FCS_TLSC_EXT.1.2(TLS) and FIA_X509_EXT.2 Test#1 (OCSP).

7.4.24 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	• Start a connection with the server using ECDHE cipher and secp256r1 curve. • Verify the connection is successful with packet capture. • Verify with logs. • Start a connection with the server using ECDHE cipher and secp384r1 curve. • Verify the connection is successful with packet capture. • Verify with logs. • Start a connection with the server using ECDHE cipher and secp521r1 curve. • Verify the connection is successful with packet capture. • Verify with logs.

Expected Test Results	• TOE must be able to establish a connection when TOE configured supported cipher and curves
Pass/Fail with Explanation	Pass. The TOE is able to establish a connection with the supported EC curves. This meets the testing requirement.

7.4.25 FCS_TLSC_EXT.1.3 Test #3

Item	Data
Test Assurance Activity	Test 3 [conditional]: 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, TOE does not implement any administrator override mechanism

7.5 UPDATE

7.5.1 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	At start-up after restart of TOE • Verify the current version on the TOE. • Check the time on TOE and do warm restart of the TOE. • Go to diag user shell and verify integrity self test was performed and passed after warm restart of TOE. • In diag user shell of TOE, run the command shown below to verify that another self-test mentioned in ST were performed by TOE on restart. • Verify via logs that warm restart of TOE was initiated.
Expected Test Results	• The TOE should execute all claimed self-tests during bootup.
Pass/Fail with Explanation	Pass. The TOE performs all claimed self-tests. This meets the testing requirements.

7.5.2 FPT_TUD_EXT.1 Test #1

Item	Data
Test Assurance Activity	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
Test Steps	• Verify current version on the TOE. • Verify TOE is setup to connect to file server securely. • Upgrade the TOE by accessing the upgrade image from file server over https port. • Verify that the update image file is getting downloaded on the TOE. • Verify that the update image file is getting installed on the TOE. • Verify that update image is installed and is also updated on the TOE. • Verify via logs TOE is updated to latest image.
Expected Test Results	• The TOE should successfully update the current version with the new version after verifying the integrity of the new image. • Evidence (e.g., screenshot or CLI output) showing old version from executing the version verification. • Evidence (e.g., screenshot or CLI output) showing new version from executing the version verification.
Pass/Fail with Explanation	Pass. The TOE software was able to be updated when an image that passes the integrity test is used. This meets the testing requirements.

7.5.3 FPT_TUD_EXT.1 Test #2 (a)

Item	Data
Test Assurance Activity	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.

Test Steps	• Check the current version on the TOE. • Open the legitimate update image file in hex editor. • Modify the legitimate image file in hex editor. • Verify modified signature file is uploaded on the file server. • Try to upgrade TOE using the modified image file, it should fail. • Verify with logs that TOE did not accept illegitimate update. • Verify that the TOE version remains the same.
Expected Test Results	• The TOE should detect and reject the modified image for a software update. • Evidence (e.g., screenshot or CLI output) showing the old version before and after the update attempt. • Logs showing the initiation and failure of the software update.
Pass/Fail with Explanation	Pass. The TOE actively rejects software updates that are corrupt. This meets the testing requirements.

7.5.4 FPT_TUD_EXT.1 Test #2 (b)

Item	Data
Test Assurance Activity	[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.

Test Steps	• Verify current version on the TOE. • Check the update image signature file in the file server. • Delete the update image signature file from the file server. • Try to upgrade the TOE using valid image but without signature file, it should fail. • Verify via logs that TOE has not accepted the image and upgrade has failed. • Verify that the TOE version remains the same.
Expected Test Results	• The TOE should detect and reject the image without signature for software update. • Logs showing the initiation and failure of the software update.
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.

7.5.5 FPT_TUD_EXT.1 Test #2 (c)

Item	Data
Test Assurance Activity	[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.
Test Steps	• Verify current version on the TOE. • Open the Signature image File in hex editor.

	• Modify signature image file in hex editor. • Verify modified signature file is uploaded on the file server. • Try to upload the TOE using valid image but with invalid signature file it should fail. • Verify via logs that has not accepted the image and upgrade is failed. • Verify that the TOE version remains the same.
Expected Test Results	• The TOE should detect and reject the image with invalid signature for software update. • Logs showing the initiation and failure of the software update.
Pass/Fail with Explanation	Pass. The TOE actively rejects software updates when an image signature is invalid. This meets the testing requirements.

7.5.6 FPT_TUD_EXT.1 Test #3 (a)

Item	Data
Test Assurance Activity	[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. 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.
Pass/Fail with Explanation	N/A, published hash is not claimed by TOE.

7.5.7 FPT_TUD_EXT.1 Test #3 (b)

Item	Data
Test Assurance Activity	[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. 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.
Pass/Fail with Explanation	N/A, published hash is not claimed by TOE.

7.6 X509-Rev

7.6.1 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	• Create a full chain of certificates to connect to the TOE. • Upload a complete certificate validation chain to the TOE. • Attempt to connect to the TOE with the complete certificate chain. • Verify the connection succeeds with packet capture. • Verify with logs.
Expected Test Results	• When a complete certificate chain is present, the TOE should establish a successful connection.
Pass/Fail with Explanation	Pass. When a complete certificate trust chain is present, the TOE is able to make a successful connection. This meets the testing requirements.

7.6.2 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	• Verify ICA certificate is missing from full chain of certificates to connect to the TOE. • Attempt to connect to the TOE with a server certificate with an incomplete chain and verify that it fails. • Verify with packet capture that the server certificate chain is incomplete. • Verify with logs.
Expected Test Results	• When an incomplete certificate chain is present, the TOE should not establish a connection.
Pass/Fail with Explanation	Pass. When an incomplete certificate trust chain is present, the TOE is not able to make a successful connection. This meets the testing requirements.

7.6.3 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	• Create a server certificate which is expired. • Attempt to connect to the TOE with the expired server certificate and verify that it fails. • Verify with packet capture. • Verify with logs.
Expected Test Results	• The TOE should deny the TLS connection when the certificate is expired.
Pass/Fail with Explanation	Pass. The TOE denied the connection because of the expired certificate. This meets the testing requirements.

7.6.4 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	1.Valid Certificate: • Create certificates with OSCP eku. CA cert: ICA cert: Server cert • Import the CA & ICA certificates on the TOE. • Configure the TOE for OCSP & Syslog • Start OCSP responder. CA Responder: ICA Responder: • Start the Syslog server using Server certificates. • Verify the logs on the TOE. • Verify successful connection with Packet Capture. ICA Status: Server status: TLS: 2.Revoked the server certificate.

	• Revoked the server certificate. • Start OCSP responder for CA and ICA certificates. CA Responder: ICA responder: • Start the Syslog server using revoked Server certificates. • Verify the logs on the TOE. • Verify the unsuccessful connection with packet capture. TLS: ICA Status: Server status: 3. Revoked Intermediate CA Certificate: • Reset the certificate chain and revoke only the intermediate CA certificate. • Start OCSP responder for CA and ICA certificates. CA Responder: ICA Responder: • Start the Syslog server using Server certificates. • Verify the logs on the TOE. • Verify the unsuccessful connection with packet capture. TLS: ICA Status:
Expected Test Results	• The TOE rejects any TLS server connection when either the intermediate certificate or the server certificate has been revoked. • The OCSP connection also shows that the certificates have been revoked. • The Packet capture depicting the specific certificate that has been revoked and the logs verifying that the TOE has denied connection by denoting that certificate has been revoked.
Pass/Fail with Explanation	Pass. Connection with revoked certificates is not accepted by the TOE which meets the requirement.

7.6.5 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	• Generate a certificate that does NOT have OCSP signing purpose. • Use this certificate in the OCSP responder. • Attempt the connection from the TOE to the TLS server and verify the connection being unsuccessful. • Verify the logs on the device. • Verify the packet capture. TLS: OCSP Responder:
Expected Test Results	• The TOE doesn't establish a TLS server connection when the OCSP signing purpose is missing and validation fails. • The packet capture shows that there is a handshake failure due to the absence of OCSP Signing. • The logs are used to validate the fact that the connection has been rejected by OCSP due to failure in certificate verification.
Pass/Fail with Explanation	Pass. The TOE does not connect to the TLS and OCSP servers if OCSP signing is missing. This meets the testing requirements.

7.6.6 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	• Attempt a connection to a remote modified TLS server using acumen-tlsc tool that would perform the necessary modification on the server certificate. Verify that the TOE rejects the connection. • Verify that the connection fails with packet capture. • Verify with the help of logs.
Expected Test Results	• The TOE denies a TLS connection when it is presented with a certificate that has been modified using the 'acumen-tlsc-v2.2e tool'. • The tool modifies the first eight bytes of the certificate. • The packet capture verifies that the connection is not established due to the bad certificate. • The logs depict that there's an encoding error thus verifying that the connection was rejected.
Pass/Fail with Explanation	Pass. The TOE rejects connections when the first 8 bytes of the certificate are modified. This meets the testing requirements.

7.6.7 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	• Attempt a connection to a remote TLS server with a modified certificate using acumen-tlsc tool and verify that it fails. • Verify with the help of packet capture. • Verify with logs.
Expected Test Results	• The TOE failing to establish a TLS connection when the last byte in the signatureValue field of the certificate is modified using the 'acumen-tlsc-v2.2e tool'. • The packet capture proving that there is a decrypt error, and the logs showing that there is a failure in establishing connection due to certificate signature failure.
Pass/Fail with Explanation	Pass. The modified certificate fails to validate. This meets the testing requirement.

7.6.8 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	• Attempt a connection to a remote TLS server using acumen-tlsc tool and modify any byte in the public key of the certificate. Verify that the connection is rejected. • Verify with packet capture. • Verify with logs.
Expected Test Results	• The TOE should reject modified certificate.
Pass/Fail with Explanation	Pass. The TOE rejects a connection when the bytes inside the public key of the server is modified. This meets the testing requirement.

7.6.9 FIA_X509_EXT.1.1/Rev Test #8a

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. **TD0527 (12/1 Update) has been applied.**
Pass/Fail with Explanation	NA, as per ST, TOE does not support EC certificate for SigGen.

7.6.10 FIA_X509_EXT.1.1/Rev Test #8b

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. **TD0527 (12/1 Update) has been applied.**
Pass/Fail with Explanation	NA, as per ST, TOE does not support EC certificate for SigGen.

7.6.11 FIA_X509_EXT.1.1/Rev Test #8c

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. **TD0527 (12/1 Update) has been applied.**
Pass/Fail with Explanation	NA, as per ST, TOE does not support EC certificate for SigGen.

7.6.12 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) (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 a CA certificate that does not contain the basic Constraints extension. • Verify the basicConstraints extension is missing on TLS server. • Attempt to load an ICA certificate without Basic Constraints extension and verify that the TOE rejects it. • Verify with logs.
Expected Test Results	• The TOE should reject certificates signed by CA that does not contain the BasicConstraints Extension.
Pass/Fail with Explanation	Pass. The TOE rejects a connection with a server if the Basic Constraints extension is missing as part of the chain. This meets the testing requirements.

7.6.13 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: 1. As part of the validation of the leaf certificate belonging to this chain; 2. 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	• Use the X509 mod tool to set basicConstraints of ICA to false. • Verify the basicConstraints flag set to FALSE on TLS server. • Attempt to load an ICA certificate with Basic Constraints extension set to FALSE and verify that the TOE rejects it. • Verify with logs.
Expected Test Results	• The TOE should reject certificates signed by CA that has CA flag set to FALSE.
Pass/Fail with Explanation	Pass. The TOE rejects a connection when the server certificate chain has Basic Constraints set to FALSE. This meets the testing requirement.

7.6.14 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. **TD0537 applied.**
Test Steps	1. Valid Certificate: • Configure the server certificate which is valid. • Configure the server certificate showing the OCSP distribution point. • Start OCSP responder. CA: ICA: • Attempt the connection from the TOE to the OCSP server and show the connection being successful. • Verify the logs on TOE. • Verify the packet capture between the TOE and the OCSP server. ICA: Server: • Verify the packet capture between the TOE and the TLS server. 2. TOE is unable to validate the certificate from the OCSP server: • Configure the server certificate. • Configure the server certificate 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 logs on TOE. • Verify the packet capture between the TOE and the TLS server. • Verify the packet capture between the TOE and the OCSP server.

Expected Test Results	• The TOE will reject the OCSP connection as the certificate used has an incorrect URL. • 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 it cannot verify via OCSP when the responder is down. This meets the testing requirements.

7.7 Crypto Test Cases

7.7.1 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, 4096 CAVP #: A3495 Pass. Based on these findings, this assurance activity is considered satisfied.

7.7.2 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 #: A3495 Pass. Based on these findings, this assurance activity is considered satisfied.

7.7.3 FCS_CKM.1 FFC

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. **TD0580 has been applied.**

Pass/Fail with Explanation	Testing for FFC Schemes using safe-prime groups is done as part of testing in CKM.2.1. Additionally, Algorithm: Safe Prime Key Generation Safe prime Groups: MODP-2048, MODP-4096 CAVP #: A3495 Pass. Based on these findings, this assurance activity is considered satisfied.

7.7.4 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	Pass. This testing was performed in conjunction with FTP_TRP.1/Admin Test #1 and FTP_ITC.1 Test #1 to demonstrate correct operation.

7.7.5 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 #: A3495 Pass. Based on these findings, this assurance activity is considered satisfied.

7.7.6 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 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.

7.7.7 FCS_COP.1/DataEncryption AES-CBC

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.
	
	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.
	
	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 Key size: 128, 256 CAVP #: A3495 Pass. Based on these findings, this assurance activity is considered satisfied.

7.7.8 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 #: A3495 Pass. Based on these findings, this assurance activity is considered satisfied.

7.7.9 FCS_COP.1/DataEncryption AES-CTR

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]. **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. **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 Key size: 128, 256 CAVP #: A3495 Pass. Based on these findings, this assurance activity is considered satisfied.

7.7.10 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 CAVP #: A3495 Pass. Based on these findings, this assurance activity is considered satisfied.

7.7.11 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, 4096 CAVP #: A3495 Pass. Based on these findings, this assurance activity is considered satisfied.

7.7.12 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 #: A3495 Pass. Based on these findings, this assurance activity is considered satisfied.

7.7.13 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 #: A3495 Pass. Based on these findings, this assurance activity is considered satisfied.

7.7.14 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 Mode: AES-256 CAVP #: A3495 Pass. Based on these findings, this assurance activity is considered satisfied.

8. Security Assurance Requirements

8.1 ADV_FSP.1 Basic Functional Specification

8.1.1 ADV_FSP.1

8.1.1.1 **ADV_FSP.1 Activity 1**

Objective	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.
Evaluator Findings	The evaluator examined 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 of the Guidance Evaluation Activities. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

8.1.1.2 **ADV_FSP.1 Activity 2**

Objective	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 examined the AGD (interface documentation) to verify that it identifies and describes the parameters for each TSFI that is identified as being security relevant. The evaluator examined the entire AGD. The evaluator verified the AGD describes the parameters for each security relevant TSFI by verifying the AGD satisfies all of the Guidance Evaluation Activities. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

8.1.1.3 **ADV_FSP.1 Activity 3**

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

Evaluator Findings	The evaluator examined the AGD (interface documentation) to develop a mapping of the interfaces to SFRs. The evaluator examined the entire AGD. Each Guidance Evaluation Activity is associated with a specific SFR. The Evaluation Findings for each Guidance Evaluation Activity identify the relevant interfaces, thus providing a mapping. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

8.2 AGD_OPE.1 Operational User Guidance

8.2.1 AGD_OPE.1

8.2.1.1 **AGD_OPE.1 Activity 1**

Objective	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 below are met by the guidance documentation. Guidance documentation shall be 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.. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

8.2.1.2 **AGD_OPE.1 Activity 2**

Objective	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 Operational guidance 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: TOE Software: Ciena SAOS 10.7.1 TOE Hardware: Ciena 3926, 3928, 3948, 5144, 5162, 5164, 5170, 5171, Large NFV Compute Server, and 8180 Service Aggregation Platforms Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

8.2.1.3 AGD_OPE.1 Activity 3

Objective	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 Operational guidance contains instructions for configuring any cryptographic engine associated with the evaluated configuration of the TOE. While performing the Guidance Evaluation Activities for the cryptographic SFRs, the evaluator ensured guidance contained the necessary instructions for configuring the cryptographic engines. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

8.2.1.4 AGD_OPE.1 Activity 4

Objective	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” & “Security Measures for the Operational Environment” specifies features that are not assessed and tested by the EAs. The evaluator ensured the Operational

	guidance makes it clear to an administrator which security functionality and interfaces have been assessed and tested by the EAs. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

8.2.1.5 **AGD_OPE.1 Activity 5** [TD0536]

Objective	In addition, the evaluator shall ensure that the following requirements are also met. a) The guidance documentation 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. b) 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: i) 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). ii) 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. c) The TOE will likely contain security functionality that does not fall in the scope of evaluation under this cPP. The guidance documentation shall make it clear to an administrator which security functionality is covered by the Evaluation Activities.
Evaluator Findings	The evaluator verified the guidance documentation contains instructions for configuring any cryptographic engines in AGD_OPE.1 Test #3. The evaluator verified the guidance documentation describes the process for verifying updates in FPT_TUD_EXT.1 Guidance 2. The evaluator verified the guidance documentation to make it clear which security functionality is covered by the Evaluation Activities in AGD_OPE.1 Test #4. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

8.3 AGD_PRE.1 Preparative Procedures

8.3.1 AGD_PRE.1

8.3.1.1 **AGD_PRE.1 Activity 1**

Objective	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).
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 "Security Measures for the Operational Environment" of the AGD. The evaluator found that these sections describe how the Operational Environment must meet:

Component	Required	Usage/Purpose Description for TOE performance
Management Workstation with SSH Client	Yes	This includes any IT Environment Management workstation with a SSH client installed that is used by the TOE administrator to support TOE administration through SSH protected channels. Any SSH client that supports SSHv2 may be used.
Local Console	Yes	This includes any IT Environment Console that is directly connected to the TOE via the Serial Console Port and is used by the TOE administrator to support TOE administration.
Audit (syslog) Server	Yes	This includes any syslog server to which the TOE would transmit syslog messages. Also referred to as audit server in the ST
Certificate Authority	Yes	This includes any Operational Environment Certificate Authority on the TOE network. This can be used to provide the TOE with a valid certificate during certificate enrollment.

Based on these findings, this assurance activity is considered satisfied.

Verdict	Pass.

8.3.1.2 AGD_PRE.1 Activity 2

Objective	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 below 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 guidance documentation describes each of the devices in the operating environment, including,

Component	Required	Usage/Purpose Description for TOE performance
Management Workstation with SSH Client	Yes	This includes any IT Environment Management workstation with a SSH client installed that is used by the TOE administrator to support TOE administration through SSH protected channels. Any SSH client that supports SSHv2 may be used.
Local Console	Yes	This includes any IT Environment Console that is directly connected to the TOE via the Serial Console Port and is used by the TOE administrator to support TOE administration.
Audit (syslog) Server	Yes	This includes any syslog server to which the TOE would transmit syslog messages. Also referred to as audit server in the ST
Certificate Authority	Yes	This includes any Operational Environment Certificate Authority on the TOE network. This can be used to provide the TOE with a valid certificate during certificate enrollment.

The section titled “Supported Hardware and Software” of AGD identifies the following supported platform:

Table 5 Supported Hardware

Hardware	Models

	Ciena Routers: Ciena 3926, 3928, 3948, 5144, 5162, 5164, 5170, 5171, Large NFV Compute Server, and 8180 Table 6 Supported Software Software: Ciena SAOS; Version: 10.7.1 Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

8.3.1.3 AGD_PRE.1 Activity 3

Objective	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 functions necessary to install and configure the TOE to work in the target operating environment, including, • Secure Installation and Configuration. • Secure Management. • Configuring Administrative Accounts and Passwords. • Configuring SSH and Console Connections. • Configuring the Remote Syslog Server. • Configuring Audit Log Options. • Configuring Event Logging. • Configuring a Secure Logging Channel. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

8.3.1.4 AGD_PRE.1 Activity 4

Objective	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. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

8.3.1.5 AGD_PRE.1 Activity 5

Objective	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 Passwords were used to determine the verdict of this work unit. The AGD describes changing the default password, the TOE provides the ability Authorized Administrators to change password and configure password policy. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

8.4 ALC Assurance Activities

8.4.1 ALC_CMC.1

8.4.1.1 **ALC_CMC.1 Activity 1**

Objective	When evaluating that the TOE has been provided and is labelled with a unique reference, the evaluator performs the work units as presented in the CEM.
Evaluator Findings	The evaluator verified that the ST, TOE and Guidance are all labeled with the same hardware versions and software. The information is specific enough to procure the TOE and it includes hardware models and software versions. The evaluator checked the TOE software version and hardware identifiers during testing by examining the actual machines used for testing. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

8.4.2 ALC_CMS.1

8.4.2.1 **ALC_CMS.1 Activity 1**

Objective	When evaluating the developer's coverage of the TOE in their CM system, the evaluator performs the work units as presented in the CEM.
Evaluator Findings	The evaluator verified that the ST, TOE and Guidance are all labeled with the same hardware versions and software. The information is specific enough to procure the TOE and it includes hardware models and software versions. The evaluator checked the TOE software version and hardware identifiers during testing by examining the actual machines used for testing. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

8.5 ATE_IND.1 Independent Testing – Conformance

8.5.1 ATE_IND.1

8.5.1.1 **ATE_IND.1 Activity 1**

Objective	The evaluator performs the CEM work units associated with the ATE_IND.1 SAR. Specific testing requirements and EAs are captured for each SFR in Sections 2, 3 and 4. The evaluator should consult Appendix 709 when determining the appropriate strategy for testing multiple variations or models of the TOE that may be under evaluation.
Evaluator Findings	The evaluator examined the TOE to determine that the test configuration is consistent with the configuration under evaluation as specified in the ST. Upon investigation, the evaluator found that each instance of the TOE used in testing was consistent with TOE description found in the Security Target. Additionally, the evaluator found that the TOE version is consistent with what was specified in the Security Target. The evaluator examined the TOE to determine that it has been installed properly and is in a known state. The details of the installed TOE and any configuration performed with the TOE are found in the separate Test Reports. The evaluator prepared a test plan that covers all of the testing actions for ATE_IND.1 in the CEM and in the SFR-related Evaluation Activities. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

8.6 AVA_VAN.1 Vulnerability Survey

8.6.1 AVA_VAN.1

8.6.1.1 **AVA_VAN.1 Activity 1 [TD0564]**

Objective	The evaluator shall document their analysis and testing of potential vulnerabilities with respect to this requirement.
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 publicly available information are provided below.

	• https://nvd.nist.gov/vuln/search • http://cve.mitre.org/cve • https://www.cvedetails.com/vulnerability-search.php • https://www.kb.cert.org/vuls/search/ • 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 • https://www.ciena.com/ The evaluator performed the public domain vulnerability searches using the following key words. The search was performed on 29 February 2024. • Ciena Switch • Ciena Router • Ciena SAOS • Ciena SAOS 10.7.1 • Ciena 3926 • Ciena 5162 • Ciena 3928 • Ciena 3948 • Ciena 5144 • Ciena 5164 • Ciena 5170 • Ciena 5171 • Ciena 8180 • Ciena • SAOS 10.7.1 • Intel XEON D1527 • Intel XEON D1539 • Intel XEON D1548 • ARM Cortex A72

	• ARM Cortex A53 • OpenSSL 3.0.8 • OpenSSH 8.4 • Linux kernel 5.4.154 • Infineon SLM9670 • Infineon SLB9665 • SAOS • Large NFV Compute Server • Linux Pluggable Authentication Module • Tls v1.2 • rsyslogd 8.1903.0 • ntp 4.2.8p15 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. Based on these findings, this assurance activity is considered satisfied.
Verdict	Pass.

8.6.1.2 AVA_VAN.1 Activity 2

Objective	Type 4 Hypotheses – Tool-Generated
	
	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	Type 4 Hypotheses – Tool-Generated 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.

9. Conclusion

The testing shows that all test cases required for conformance have passed testing.

End of Document