Assurance Activity Report for
Progress LoadMaster

Progress LoadMaster Security Target

Version 1.7

collaborative Protection Profile for Network Devices
Version 3.0e

AAR Version 0.5, June 18, 2026

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:
Progress Software Corporation

The Author of the Security Target:
Intertek Acumen Security

The TOE Evaluation was Sponsored by:
Progress Software Corporation

Evaluation Personnel:
Rupal Gupta
Theo Ajibade
Akshay Jain

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	May 05, 2025	Initial Release
0.2	Oct 15, 2025	Removal of tls 1.3
0.3	Feb, 17,2026	Ready for Peer Lead Review
0.4	May,06, 2026	Checkout Submission
0.5	June 18, 2026	Updates based on ECR comments

CONTENTS

1 TOE OVERVIEW

The TOE is the Progress Software Corporation's Progress LoadMaster X25-NG, X40-NG, ECS CM H2 NG and ECS CM H3 NG running on LoadMaster OS 7.2.54.18. The LoadMaster simplifies the management of networked resources, and optimizes and accelerates user access to diverse servers, content, and transaction-based systems. The TOE is comprised of hardware and software and represents a complete network device providing load balancing functionality.

2 ASSURANCE ACTIVITIES IDENTIFICATION

The Assurance Activities contained within this document include all those defined within the NDcPP 3.0e based upon the core SFRs and those implemented based on selections within the PP.

3 TEST EQUIVALENCY JUSTIFICATION

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 NDcPPv3.0e. Additionally, a comparison of the data presented in section 3 is provided to identify a testing subset that will exercise each of the differences in TOE models.

3.1 PLATFORM/HARDWARE DEPENDENCIES

The TOE boundary is inclusive of all hardware required by the TOE. The hardware platforms do not provide any of the TSF functionality. The hardware within the TOE only differs by configuration and performance.

Table 1 – TOE Models

Model	LoadMaster X25-NG	LoadMaster X40-NG	ECS CM H2 NG	ECS CM H3 NG	Virtual LoadMaster
Processor	Intel Xeon Silver 4316 (Ice Lake)	Intel Xeon Silver 4316 (Ice Lake)	Intel Xeon Silver 4316 (Ice Lake)	Intel Xeon Silver 4316 (Ice Lake)	Intel Xeon Gold 5222 (Cascade Lake)
RAM	64 GB RAM (evaluated)	64 GB RAM	128 GB RAM	128 GB RAM	4GB (evaluated)
Network	2 1Gb Ethernet 8 10Gb Ethernet Fiber	2 1Gb Ethernet 8 10Gb Ethernet Fiber	2 1Gb Ethernet 8 10Gb Ethernet Fiber 4 25Gb Ethernet Fiber 4 100Gb Ethernet Fiber	2 1Gb Ethernet 8 10Gb Ethernet Fiber 4 25Gb Ethernet Fiber 4 100Gb Ethernet Fiber	2 10Gb virtual NIC (evaluated)
Platform	LoadMaster OS 7.2.54.18	LoadMaster OS 7.2.54.18	LoadMaster OS 7.2.54.18	LoadMaster OS 7.2.54.18	LoadMaster OS 7.2.54.18 on ESXi v7.0 U3

The TOE chassis includes varying form factors. Although the chassis may differ, it does not affect the functionality of the TOE.

Result: All platforms are equivalent.

3.2 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 models are equivalent.

3.3 SOFTWARE/OS DEPENDENCIES:

The underlying OS is installed with the application-level software on each of the devices. The LMOS software has version 7.2.54.18. The LMOS software can run on a dedicated hardware appliances and on servers running on ESXi 7.0 U3 hypervisor. All models include the same security functionality. There are no specific dependencies on the OS apart from the additional hypervisor layer in case of virtual model.

Table 2– TOE OS Comparison

TOE Model	Description	Analysis
Operating System – This is the OS that runs on the platform		
LoadMaster X25-NG	LoadMaster OS 7.2.54.18	All four hardware devices are running the same OS and installed using the same image. The virtual TOE is also running the same OS but on ESXi v7.0 U3 hypervisor. Verdict: The Four hardware models are equivalent.
LoadMaster X40-NG	LoadMaster OS 7.2.54.18	
ECS CM H2 NG	LoadMaster OS 7.2.54.18	
ECS CM H3 NG	LoadMaster OS 7.2.54.18	
Virtual LoadMaster	LoadMaster OS 7.2.54.18 on ESXi v7.0 U3	

Result :

- All Hardware models are equivalent. One of the four hardware models will be tested.
- One virtual model will be tested.

3.4 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. Of note, the TOE uses the same cryptographic module to provide its cryptographic functionality. This is the same across platforms.

Result :

- There are no differences in the included libraries.
- All models are equivalent.

3.5 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 for each of these devices. Each device runs the same version of LoadMaster OS. 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 models are equivalent.

3.6 DIFFERENCE COMPARISON

All platforms run the same software and perform identical functionality. All hardware platforms use identical microarchitecture processor. The virtual model uses a different microarchitecture processor.

3.7 TLS EQUIVALENCY JUSTIFICATION

TOE establishes two distinct cryptographic communication channels with trusted IT products:

TLS v1.2 client connections with TLS client authentication: Used for interactions with Syslog servers.

TLS v1.2 client connections with TLS client authentication: Utilized for communication with Lightweight Directory Access Protocol/Active Directory (LDAP/AD) servers.

The TOE leverages a unified TLS implementation via the OpenSSL cryptographic library (version 1.1.1u), which is identically configured and executed across all TOE models. A global TLS configuration governs each trusted channel, ensuring uniform enforcement of cryptographic policies and operational consistency regardless of the TOE model. To ensure comprehensive coverage, TLSC and X.509 testing will be conducted against a LDAP and a Syslog. The TOE models exhibit identical behavior, and testing is equivalent across all models.

Therefore, the testing will be split by performing complete TLSC and X509 testing against one TLS server per model as follows:

Table 3- TLSC Tested Subset details

TOE's model	TLS trusted channel fully tested
LoadMaster X25-NG	Syslog over TLS v1.2 client connection with x509 authentication

Virtual LoadMaster running on ESXi v7.0 U3	LDAP over TLS v1.2 client connection with x509 authentication

3.8 RECOMMENDATIONS/CONCLUSIONS

Based on the equivalency rationale listed above, testing will be performed on the following subset:

Table 4: Tested Subset detail

Platform to be tested	CPU	TLSC testing coverage	Remaining Testing Coverage
LoadMaster X25-NG	Intel Xeon Silver 4316 (Ice Lake)	Syslog over TLS trusted channel	Testing activities for NDcPP v3.0E (excluding TLSC testing), will be fully conducted on the listed appliances, running LoadMaster OS 7.2.54.18 Additionally, TLSC testing will be performed as specified in Table 3.
Virtual LoadMaster running on ESXi v7.0 U3	Intel Xeon Gold 5222 (Cascade Lake)	LDAP over TLS trusted channel	

4 TEST BED DESCRIPTIONS

4.1 TEST BED

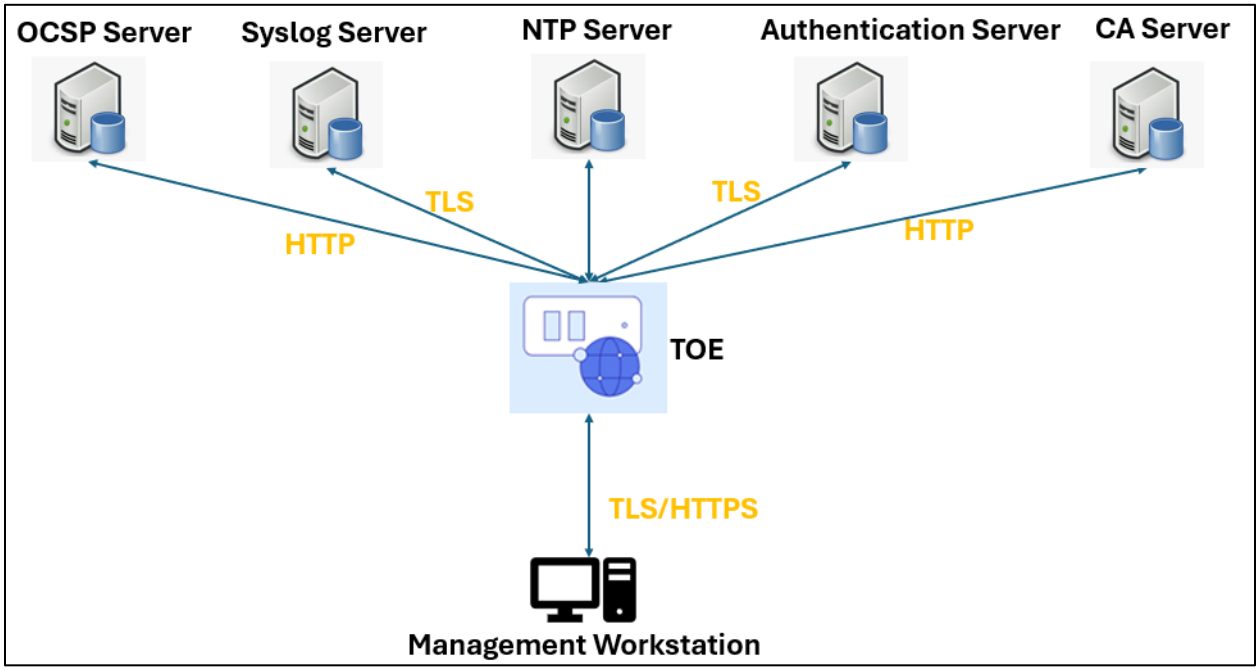


Figure 1 – Physical TOE Management Interfaces

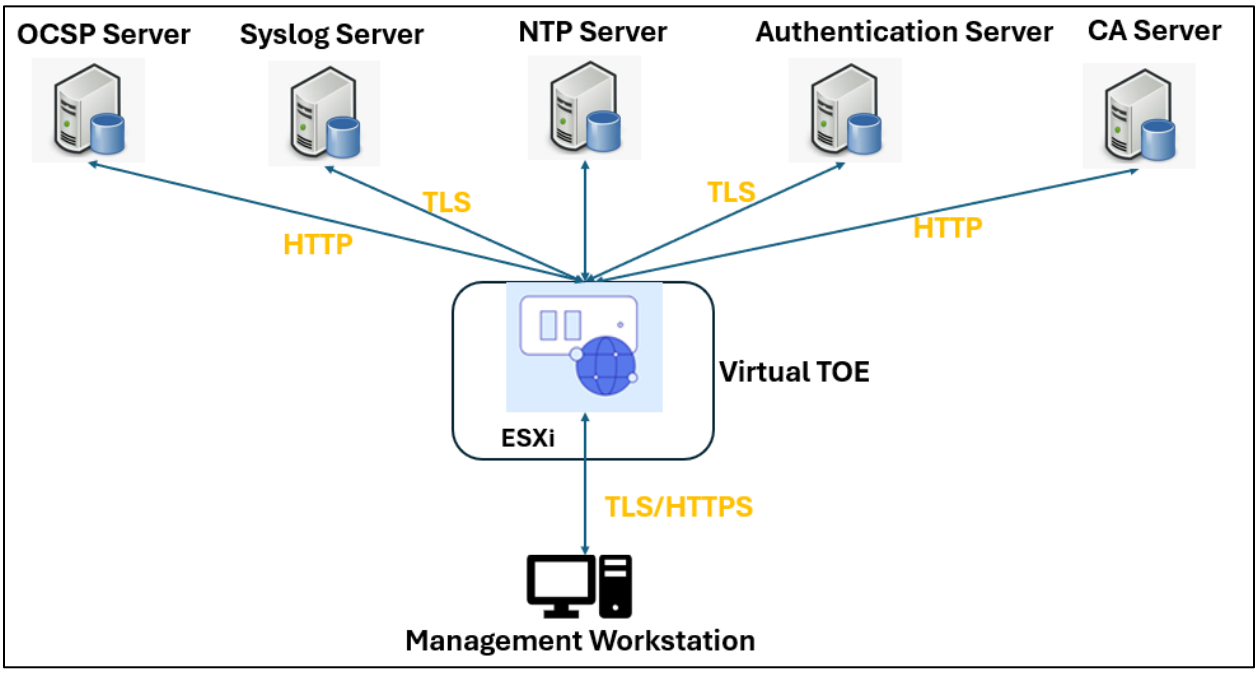


Figure 2 – Virtual TOE Management Interfaces

4.2 CONFIGURATION INFORMATION

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

Table 5 - Physical LoadMaster Configuration

Device Details		System Details				
Device Name	**Function**	**Protocols**	**OS, including version**	**Timing Source**	**Software & Tools, including version**	**Credentials**
LoadMaster X25-NG	TOE	TLS/HTTPS	LoadMaster OS v7.2.54.18	Manually set and verified	N/A	bal/123TesT321
Tripplite B096-048	Console	SSH	N/A	NTP	N/A	acumensec/123TesT321
Test User Laptop	Test User Laptop	SSH, TLS/HTTPS	22.04.5 LTS (GNU)	NTP	Wireshark v4.4.7, winscp, XCA v2.1.1, MobaXterm 25.2 build 5296, HxD Hex Editor version 2.5.0.0	N/A

Cisco Catalyst 2960-L	Switch	N/A	IOS-XE	NTP	N/A	N/A
Test VM1	Syslog Server/NTP Server 1/CA Server	Syslog, TLS,HTTP,NTP	22.04.5 LTS (GNU)	NTP	Rsyslog v8.2112 Ntpd v 4.2.8p15 Wireshark 3.6.2 XCA v2.4.0 OpenSSL 3.0.2 15 OpenSSH_8.9p1 acumen-tls version 4.2.1	acumensec/123TesT321
Test VM2	LDAP Server/ NTP Server 2/ CA Server / OCSP Server	LDAP, TLS,HTTP,NTP	22.04.5 LTS (GNU)	NTP	Ntpd v 4.2.8p15 Apache v2.4.52 openldap v2.5.19 Wireshark 3.6.2 acumen-tls version 4.2.1 XCA v2.4.0 OpenSSL 3.0.2 15 OpenSSH_8.9p1	acumensec/123TesT321
Test VM3	NTP Server 3	NTP, TLS	22.04.5 LTS (GNU)	NTP	Ntpd v 4.2.8p15 Wireshark 3.6.2 XCA v2.4.0 OpenSSL 3.0.2 15 OpenSSH_8.9p1 acumen-tls version 4.2.1	acumensec/123TesT321
Test VM4	NTP Server 4	NTP, TLS	22.04.5 LTS (GNU)	NTP	Ntpd v 4.2.8p15 Wireshark 3.6.2 XCA v2.4.0 OpenSSL 3.0.2 15 OpenSSH_8.9p1 acumen-tls version 4.2.1	acumensec/123TesT321

Meraki VPN	Gateway (Also acts like a router)	N/A	IOS-XE	N/A	N/A	N/A

Table 6 - Virtual LoadMaster Configuration

Device Details		System Details				
Device Name	**Function**	**Protocols**	**OS, including version**	**Timing Source**	**Software & Tools, including version**	**Credentials**
Virtual LoadMaster	TOE	TLS/HTTPS	LoadMaster OS v7.2.54.18	Manually set and verified	N/A	bal/123TesT321
VMware EXSI hypervisor	Console	N/A	ESXi v7.0 U3	Manually set and verified	N/A	acumensec/123TesT321
Test User Laptop	Test User Laptop	SSH, HTTPS	22.04.5 LTS (GNU)	NTP	Wireshark v4.4.7, winscp, XCA v2.4.0, MobaXterm 25.2 build 5296, HxD Hex Editor version 2.5.0.0	N/A
Cisco Catalyst 2960-L	Switch	N/A	IOS-XE	NTP	N/A	N/A
Test VM1	Syslog Server/NTP Server 1/CA Server	Syslog, TLS,HTTP,NTP	22.04.5 LTS (GNU)	NTP	Rsyslog v8.2112 Ntpd v 4.2.8p15 Wireshark 3.6.2 XCA v2.4.0 OpenSSL 3.0.2 15 OpenSSH_8.9p1 acumen-tls version 4.2.1	acumensec/123TesT321
Test VM2	LDAP Server/ NTP Server 2/ CA Server / OCSP Server	LDAP, TLS,HTTP,NTP	22.04.5 LTS (GNU)	NTP	Ntpd v 4.2.8p15 Apache v2.4.52 openldap v2.5.19 Wireshark 3.6.2 acumen-tls version 4.2.1	acumensec/123TesT321

					XCA v2.4.0 OpenSSL 3.0.2 15 OpenSSH_8.9p1	
Test VM3	NTP Server 3	NTP,TLS	22.04.5 LTS (GNU)	NTP	Ntpd v 4.2.8p15 Wireshark 3.6.2 XCA v2.4.0 OpenSSL 3.0.2 15 OpenSSH_8.9p1 acumen-tls version 4.2.1	acumensec/123TesT321
Test VM4	NTP Server 4	NTP, TLS	22.04.5 LTS (GNU)	NTP	Ntpd v 4.2.8p15 Wireshark 3.6.2 XCA v2.4.0 OpenSSL 3.0.2 15 OpenSSH_8.9p1 acumen-tls version 4.2.1	acumensec/123TesT321
Meraki VPN	Gateway (Also acts like a router)	N/A	IOS-XE	N/A	N/A	N/A

4.3 TEST TIME AND LOCATION

All testing was carried out at Acumen Security office located at 2400 Research Blvd Suite #395, Rockville, MD 20850. Testing occurred from May 2025 to May 2026.

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

4.4 REGRESSION TESTING

Regression testing was performed for 7.2.54.18 due to the following reasons:

- Security updates for LoadMaster Security Vulnerabilites: CVE-2026-8037/ CVE-2026-33691

The evaluator performed regression testing on the following test cases:

- FPT_TUD_EXT.1 TEST #1

Regression testing was performed for 7.2.54.17 due to the following reasons:

- Security updates for LoadMaster Security Vulnerabilites: CVE-2026-3517 / CVE-2026-3518 / CVE-2026-3519 / CVE-2026-4048 / CVE-2026-21876

The evaluator performed regression testing on the following test cases:

- FPT_TUD_EXT.1 TEST #1

Regression testing was performed for 7.2.54.16 due to the following reasons:

- No log displayed for OCSP Server for LDAP connection
- Unsupported SignatureSchemes in the Client hello with respect to claimed cipher suites.

The evaluator performed regression testing on the following test cases:

- FCS_TLSC_EXT.1.1 TEST #2
- FCS_TLSC_EXT.1.1 TEST #4B
- FCS_TLSS_EXT.1.1 TEST #1
- FCS_TLSS_EXT.1.1 TEST #2A
- FCS_TLSS_EXT.1.1 TEST #2B
- FPT_TUD_EXT.1 TEST #1
- FIA_X509_EXT.1.1/REV TEST #2
- FIA_UIA_EXT.1 TEST #1

5 DETAILED TEST CASES (TSS AND AGD ACTIVITIES)

5.1 MANDATORY REQUIREMENTS

5.1.1 SECURITY AUDIT (FAU)

5.1.1.1 FAU_GEN.1 AUDIT DATA GENERATION

5.1.1.1.1 FAU_GEN.1 TSS

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

Evaluator Findings:

The evaluator examined the TSS and ensured that it identifies what information is logged to identify the relevant cryptographic key during generating/import, changing, or deleting.
The relevant information is found in the following section(s): **FAU_GEN.1 AUDIT DATA GENERATION (CPP_ND_V3.0E) AND FAU_GEN.2 USER IDENTITY ASSOCIATION (CPP_ND_V3.0E)**
Upon investigation, the evaluator found that the TSS states that: **The TSF generates audit records for the auditable events specified in Table 10 as well as the following:**
• **Startup and shut-down of the audit function**
• **Generation/importing, changing, or deleting certificates and cryptographic keys. The TSF identifies the certificate or key being operated on by including the following in the audit record:**
  ○ **Generated Keys:**
  ▪ **CSRs: X.509 Subject associated with the key**
  ▪ **Self-Signed Cert: hostname (in the CN)**
• **Uploaded/Imported Keys: uploaded filename or certificate name.**
• **For changing, and deleting of certificates and associated keys, the TOE logs the certificate name.**
Each of the events is specified in the audit record is in enough detail to identify the user for which the event is associated, when the event occurred, where the event occurred, the outcome of the event, and the type of event that occurred.

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

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

The evaluator shall ensure that this mapping of audit events to TOE components accounts for, and is consistent with, information provided in Table 1, as well as events in Tables 2, 4, and 5 (where applicable to the overall TOE). This includes that the evaluator shall confirm that all components defined as generating audit information for a particular SFR should also contribute to that SFR as defined in the

mapping of SFRs to TOE components, and that the audit records generated by each component cover all the SFRs that it implements.

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

Verdict:
PASS.

5.1.1.1.2 FAU_GEN.1 AGD

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

Evaluator Findings:
The evaluator checked the guidance documentation and ensured that it provides an example of each auditable event required by FAU_GEN.1 (i.e. at least one instance of each auditable event, comprising the mandatory, optional and selection-based SFR sections as applicable, was provided from the actual audit record). Upon investigation, the evaluator found that the AGD section **Sample Audit Logs** contains a listing and description of each of the fields in generated audit records that contain the information required in FAU_GEN.1.2, as well as an example audit record. The evaluator next compared this list of events to the auditable events listed in the NDcPP and found that they match.

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

Evaluator Findings:

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

Upon investigation, the evaluator determined that the following **administrative actions that are related to TSF data were described and configuration steps were given on how to accomplish them either using WebUI and/or CLI console interface:**

Administrative Activity	**Method (Command/GUI Configuration)**	**Section**
Audit configuration	Graphical User Interface	Section Titled: **'Secure Remote logging'**
User Creation	Graphical User Interface	Section Titled: **'User Creation'**
Software update	Graphical User Interface	Section Titled: **'Installing an Update Image'**
Setting time	Graphical User Interface	Section Titled: **'Setting Date/Time'**
Configuring banner	Graphical User Interface	Section Titled: **'Set the WUI Banner'** **'Set the CLI Banner'**

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	**Method (Command/GUI Configuration)**	**Test Case(s)**
Audit configuration	Graphical User Interface	FAU_STG_EXT.1 Test #1
User Creation	Graphical User Interface	FIA_PMG_EXT.1 Test#1
Software update	Graphical User Interface	FPT_TUD_EXT.1 Test #1
Setting time	Graphical User Interface	FPT_STM_EXT.1 Test #1
Configuring banner	Graphical User Interface	FTA_TAB.1 Test#1

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

Evaluator Findings:		
The evaluator examined the guidance documentation and made a determination of which administrative commands, including subcommands, scripts, and configuration files, are related to the configuration (including enabling or disabling) of the mechanisms implemented in the TOE that are necessary to enforce the requirements specified in the cPP. The following table summarize the evaluator's activity outcome:		
Administrative Activity	**Method (Command/GUI Configuration)**	**Section**
Login and Logout	**Logging to Local Console:** • You will need a PC to connect via COM+ (Console) port with a terminal emulation application, or a standard VGA and keyboard. Use a null modem cable (reversal) to connect the COM+ port to the LoadMaster COM port on the rear of the unit. • The COM+ settings should be 115200, 8, N,1. • After initially deploying and powering on a LoadMaster, ideally the IP address of the LoadMaster will be obtained via DHCP. • If the IP address has not been obtained via DHCP, or if the address details of the LoadMaster need to be changed, the console can be used to configure the IP address of the LoadMaster, the default gateway address and the name server addresses. • To go through this menu, simply log in to the console using the default username and password (bal and 1fourall). **Logout of Local Console:**	• Login Via WUI • Login Via Console • Logging Out

	• The console main menu appears whenever the console is accessed. • Press q to Quit the console session. **Logging in to Web Interface:** • The following steps describe the login process via the WUI: • Launch a web browser from a laptop that is connected to the device. • Log in to the WUI via HTTPS using the IP address assigned during installation, the 'bal' administrative login, and the password you specified during installation. • Download the LoadMaster issuing CA ECDSA certificate and install it in the management workstation certificate store and/or the browser certificate store. • After initially logging in to the LoadMaster, if Session Management is enabled - some login information is displayed: • The last login time and IP address of the current user • The number of successful logins by the current user in the last 30 days • The total number of failed logins attempts by any user (including unknown usernames) since the last successful login • Audit Logs are generated for each action which is performed by a user; either using the API or the WUI.	

	• The primary user (bal) always has full permissions. Secondary users may be restricted to certain functions. • Only the bal user is permitted to set the Basic Authentication password. • No administrative or security-relevant functions are available until the user has successfully authenticated, and the feedback during authentication is obscured. • Only authenticated users with the appropriate permissions can view or modify TSF data. Unauthorized users cannot perform any security-related actions. • Users must be authenticated before logging on to the LoadMaster. • Users with the 'All Permissions' permission set can view the Enable Session Management, Require Basic Authentication, and the Basic Authentication Password fields. However, users with the 'All Permissions' permission set can configure the Failed Login Attempts and Idle Session Timeout values. • Users with the 'User Administration' permissions set can view the screen but all buttons and input fields are greyed out. • All other users cannot view the WUI Session Management, Currently Active Users or Currently Blocked Users sections of the WUI Configuration screen.	

	Logging out of Web Interface: • After logging in via the WUI, the user can log out by clicking the Logout button, located in the **top right-hand corner** of the screen .	

Resetting passwords	**CHANGING THE PASSWORD FOR CURRENT USER**
	To change the password for the currently logged-in user via the WUI:
	1. Log in to the Web User Interface using valid credentials.
	2. Navigate to System Configuration > System Administration > User Management > Change Password.
	3. Enter the following details:
	  ○ Current Password
	  ○ New Password
	  ○ Re-enter New Password
	4. Click on Set Password to apply the changes.
	
	Note: Ensure the new password meets any configured password policy requirements.
	Changing The Password For Another User
	To change the password of a different user (not the currently logged-in user):
	1. Navigate to System Configuration > System Administration > User Management > Local Users.
	2. Select the target user from the list.
		• Change The Password

	3. Click on Modify. 4. Enter and confirm the new password, then save the changes. Note: Only users with appropriate administrative privileges can modify other users' passwords.	
Create CSR	• In the main menu, go to Certificates & Security > Generate CSR. • The following are the instructions for establishing the above fields in the CSR: ○ Letter Country Code (ex. US) ○ The 2-letter country code that should be included in the certificate, for example US should be entered for the United States. ○ State/Province (Entire Name – New York, not NY) ○ The state which should be included in the certificate. Enter the full name here, for example New York, not NY.	• Generate CSR (CERTIFICATE SIGNING REQUEST)

	○ City ○ The name of the city that should be included in the certificate. ○ Company (Organization) ○ The name of the organization or company which should be included in the certificate. ○ Organization Unit (e.g., Marketing, Finance, Sales) ○ The department or organizational unit that should be included in the certificate. ○ Common Name ○ The Fully Qualified Domain Name (FQDN) for your web server. ○ Email Address ○ The email address of the responsible person or organization that should be contacted regarding this certificate. ○ SAN/UCC Names	

	○ A space-separated list of alternate names. • Fill in the details in the resulting screen. The Common Name field is mandatory, all other fields are optional. • Click Create CSR. • Alter clicking the Create CSR button, the following screen appears: • The top part of the screen should be copied and pasted into a plain text file and sent to the Certificate Authority of your choice. They will validate the information and return a validated certificate. • The lower part of the screen is your private key and should be kept in a safe place. This key should not be disseminated as you will need it to use the certificate. Copy and paste the private key into a plain text file (do not use an application such as Microsoft Word) and keep the file safe. • The TSF shall validate the chain of certificates from the Root CA upon receiving the CA Certificate Response.	

Import Trusted CA Certificate	• To upload Root and Intermediate CA Certificates, before enabling certificate-based admin WUI access: • Navigate to Certificates & Security → Intermediate Certificates. • Upload the issuing CA and Root CA certificates required to validate administrator certificates for client authentication.	• Intermediate Certificate
Upgrading Firmware	To install an update image on LoadMaster: • Unzip the archive received from Progress on a laptop or other device with a browser. ○ In it are the update image and an XML file. • Open the LoadMaster WUI and navigate to System Configuration > System Administration > Update Software. • Click on the Browse button next to Software Update File and select the update image from Step 1. • Click on the Browse button next to Verification File and select the XML file from Step 1. • Click Update Machine. After the system validates the image, you'll be asked to confirm the installation to continue. • Once the update is completed, the system asks you to confirm rebooting the system. • The current version of the TOE is displayed in the top-right corner of the Home page screen.	• Installing an Update image

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

Evaluator Findings:

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

The following table maps the management function claimed by ST to the corresponding AGD section:

Management Function	AGD Sections
Ability to administer the TOE remotely;	LOGIN VIA WUI
Ability to configure the access banner;	SET THE WUI BANNER and SET THE CLI BANNER
Ability to configure the remote session inactivity time before session termination;	IDLE SESSION TIMEOUT
Ability to update the TOE, and to verify the updates using digital signature capability prior to installing those updates;	INSTALLING AN UPDATE IMAGE
Ability to start and stop services;	START/STOP WEBUI ACCESS MANAGING SYSLOG SERVICE
Ability to modify the behaviour of the transmission of audit data to an external IT entity;	GENERATE CSR (CERTIFICATE SIGNING REQUEST), TLS CIPHER SUITES AND ROLE, and SECURE REMOTE LOGGING
Ability to configure the list of TOE-provided services available before an entity is identified and authenticated, as specified in FIA_UIA_EXT.1;	SET THE WUI BANNER and SET THE CLI BANNER INSTALLING AN UPDATE IMAGE
Ability to manage the cryptographic keys;	GENERATE CSR (CERTIFICATE SIGNING REQUEST)
Ability to configure the cryptographic functionality;	GENERATE CSR (CERTIFICATE SIGNING REQUEST)
Ability to configure the list of supported (D)TLS ciphers;	ENABLING AND CONFIGURING TLS CIPHER SUITES FOR SECURE COMMUNICATION
Ability to re-enable an Administrator account;	FAILED LOGIN ATTEMPTS
Ability to set the time which is used for time-stamps;	SETTING DATE/TIME
Ability to configure NTP;	CONFIGURING NTP SERVER
Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors;	INTERMEDIATE CERTIFICATE
Ability to generate Certificate Signing Request (CSR) and process CA certificate response;	GENERATE CSR (CERTIFICATE SIGNING REQUEST), INTERMEDIATE CERTIFICATE
Ability to administer the TOE locally;	LOGIN VIA CONSOLE
Ability to configure the local session inactivity time before session termination or locking;	IDLE SESSION TIMEOUT
Ability to configure the authentication failure parameters for FIA_AFL.1].	FAILED LOGIN ATTEMPTS

Verdict:
PASS.

5.1.1.2 FAU_GEN.2 USER IDENTITY ASSOCIATION

5.1.1.2.1 FAU_GEN.2 TSS

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

5.1.1.2.2 FAU_GEN.2 AGD

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

5.1.1.3 FAU_STG_EXT.1 PROTECTED AUDIT EVENT STORAGE

5.1.1.3.1 FAU_STG_EXT.1 TSS

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

Evaluator Findings:
The evaluator examined the TSS and ensured that it describes the means by which the audit data are transferred to the external audit server, and how the trusted channel is provided. The relevant information is found in the following section(s): **FAU_STG_EXT.1 PROTECTED AUDIT EVENT STORAGE (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF transmits generated audit data to an external server using the TLS trusted channel specified in FTP_ITC.1. The TSF sends audit records to the external server as the audit records are generated. The TSF does not retransmit audit logs that were generated while the connection to the audit server was down but maintains a log file of audit records locally in the event of an interruption of communication with the remote audit server.**

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

Evaluator Findings:
The TOE is not a distributed TOE. The TSS section **FAU_STG_EXT.1 PROTECTED AUDIT EVENT STORAGE (CPP_ND_V3.0E)** states that: **The TOE is a standalone TOE that stores audit data locally.**

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

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

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

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

The evaluator shall examine the TSS to ensure that it details whether the transmission of audit data to an external IT entity can be done in real-time, periodically, or both. In the case where the TOE is capable of performing transmission periodically, the evaluator shall verify that the TSS provides details about what event stimulates the transmission to be made as well as the possible acceptable frequency for the transfer of audit data.

Evaluator Findings:
The evaluator examined the TSS and ensured that it details whether the transmission of audit information to an external IT entity can be done in real-time, periodically, or both. In the case where the TOE is capable of performing transmission periodically, the evaluator verified that the TSS provides details about what event stimulates the transmission to be made as well as the possible acceptable frequency for the transfer of audit data. The relevant information is found in the following section(s): TOE Summary Specification **FAU_STG_EXT.1 PROTECTED AUDIT EVENT STORAGE (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF sends audit records to the external server as the audit records are generated. The TSF does not retransmit audit logs that were generated while the connection to the audit server was down but maintains a log file of audit records locally in the event of an interruption of communication with the remote audit server.**

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

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

The evaluator shall examine the TSS to ensure it describes the amount of audit data that can be stored locally and how these records are protected against unauthorized modification or deletion.

Evaluator Findings:
The evaluator examined the TSS and ensured it describes the amount of audit data that are stored locally and how these records are protected against unauthorized modification or deletion. The relevant information is found in the following section(s): TOE Summary Specification **FAU_STG_EXT.1 PROTECTED AUDIT EVENT STORAGE (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF allows all available space to be used to store audit records. For the Virtual Load Master that is up to 7GB. The HW Load Master models have up to 25GB of space available to store audit records. Only Authorized Administrators can access the audit events and have the ability to clear the audit events. The TOE has a capability of deleting the logs.**

The evaluator shall examine the TSS to ensure it describes the method implemented for local logging, including format (e.g. buffer, log file, database) and whether the logs are persistent or non-persistent.

Evaluator Findings:
The evaluator examined the TSS and ensured it describes the method implemented for local logging, including format (e.g. buffer, log file, database) and whether the logs are persistent or non-persistent. The relevant information is found in the following section(s): TOE Summary Specification **FAU_STG_EXT.1 PROTECTED AUDIT EVENT STORAGE (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TOE stores persistent audit records in a log file locally, ensuring that audit records are preserved across reboots or power cycles.**

The evaluator shall examine the TSS to ensure it describes the conditions that must be met for authorized deletion of audit records.

Evaluator Findings:
The evaluator examined the TSS and ensured it describes the conditions that must be met for authorized deletion of audit records. The relevant information is found in the following section(s): TOE Summary Specification **FAU_STG_EXT.1 PROTECTED AUDIT EVENT STORAGE (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **Only Authorized Administrators can access the audit events and have the ability to clear the audit events. The TOE has a capability of deleting the logs.**

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

Evaluator Findings:
The evaluator examined the TSS and ensured that it details the behaviour of the TOE when the storage space for audit data is full. When the option 'overwrite previous audit record' is selected this description should include an outline of the rule for overwriting audit data. If 'other actions' are chosen such as sending the new audit data to an external IT entity, then the related behaviour of the TOE is detailed in the TSS. The relevant information is found in the following section(s): TOE Summary Specification **FAU_STG_EXT.1 PROTECTED AUDIT EVENT STORAGE (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **When local audit storage is exhausted, the TSF will not record new events locally until additional space is available.** **The TSF drops new audit data when the local storage space for audit data is full. Records are still transmitted to the remote syslog endpoint, and so the audit trail is preserved. Local audit records will be temporarily stored in a buffer until they can be stored. The TSF allows all available space to be used to store audit records. Warning is issued by the TOE before the local storage space for audit data is full.**

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

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

Verdict:

PASS.

5.1.1.3.2 FAU_STG_EXT.1 AGD

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

Evaluator Findings:
The evaluator examined the guidance documentation and ensured it describes how to establish the trusted channel to the audit server, as well as describe any requirements on the audit server (particular audit server protocol, version of the protocol required, etc.), as well as configuration of the TOE needed to communicate with the audit server. The relevant information is found in the following section(s): **SECURE REMOTE LOGGING** Upon investigation, the evaluator found that the AGD states that: **The LoadMaster can produce various warning and error messages using the syslog protocol. These messages are normally stored locally. Syslog messages are transmitted securely using TLS to remote servers. The LoadMaster uses OCSP to check the validity of the server certificates supplied by configured syslog servers. If these checks fail, connections to the server are not permitted.** 1. **In the left frame menu, click System Configuration > System Log Files > Syslog Options.** 2. **By entering the relevant IP address in the Syslog host text box and select the severity and click Add Syslog Host.** 3. **Also enter the Syslog Port enter *any port other than 601* and click on Set Port.** 4. **Select the Syslog Protocol either TCP, UDP or TLS.** 5. **Enable the Server Certificate Validation.** **Note: secure syslog channel is restricted to TLSv1.2**

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

Evaluator Findings:
The evaluator also examined the guidance documentation and determined that it describes the relationship between the local audit data and the audit data that are sent to the audit log server. The relevant information is found in the following section(s): **AUDITING** Upon investigation, the evaluator found that the AGD states that: **Progress LoadMaster sends audit records to the external server as the audit records are generated. It transmits generated audit data to an external server using the TLS trusted channel. It does not retransmit audit logs that were generated while the connection to the audit server was down.** **When local audit storage is exhausted, the TSF will not record new events locally until additional space is available. Records are still transmitted to the external server, and so the audit trail is preserved. Local audit records will be temporarily stored in a buffer until they can be stored. The TOE stores persistent audit records locally with a minimum storage size of [7GB], ensuring that audit records are preserved across reboots or power cycles. TSF allows all available space to be used to store audit records. For the Virtual Load Master that is up to 7GB. The HW Load Master models have up to 25GB of space. Note: The audit log storage size is not configurable. The TOE restricts the ability to determine and modify the behavior of transmission of audit data to an external IT entity to Security Administrators. The TOE is a single standalone component that stores audit data locally.**

The evaluator shall examine the guidance documentation to ensure it describes any configuration required for protection of the locally stored audit data against unauthorized modification or deletion.

Evaluator Findings:
The evaluator also examined the guidance documentation and iit describes any configuration required for protection of the locally stored audit data against unauthorized modification or deletion. The relevant information is found in the following section(s): **AUDITING** Upon investigation, the evaluator found that the AGD states that: **The TSF protects audit data from unauthorized modification and deletion through the restrictive administrative interfaces. This protection is enforced by default and does not require any additional configuration.**

If the storage size is configurable, the evaluator shall review the Guidance Documentation to ensure it contains instructions on specifying the required parameters.

Evaluator Findings:
NA. the storage size is not configurable.

If more than one selection is made for FAU_STG_EXT.1.5, the evaluator shall review the Guidance Documentation to ensure it contains instructions on specifying which action is performed when the local storage space is full.

Evaluator Findings:
N/A. This assurance activity is not applicable.

Verdict:

PASS.

5.1.2 CRYPTOGRAPHIC SUPPORT (FCS)

5.1.2.1 FCS_CKM.1 CRYPTOGRAPHIC KEY GENERATION

5.1.2.1.1 FCS_CKM.1 TSS

The evaluator shall ensure that the TSS identifies the key sizes supported by the TOE. 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 ensured that the TSS identifies the key sizes supported by the TOE. The relevant information is found in the following section(s): TOE Summary Specification **FCS_CKM.1 CRYPTOGRAPHIC KEY GENERATION (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF generates P-256 ECDSA keys for the administrative UI (HTTPS). The TSF generates and verifies P-256, P-384, and P-521 ECDH for the ECDHE key establishment used in TLS. These Keys are generated as per FIPS PUB 186-5, "Digital Signature Standard (DSS)", Appendix A.2.The relevant NIST CAVP certificate is listed in** Error! Reference source not found.**.**

Verdict:

PASS.

5.1.2.1.2 FCS_CKM.1 AGD

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

Evaluator Findings:
The evaluator verified that the AGD guidance instruct the administrator how to configure the TOE to use the selected key establishment scheme(s). The relevant information is found in the following section(s): **USE OF CRYPTOGRAPHIC KEY GENERATION SCHEMES** Upon investigation, the evaluator found that the AGD states that: **No manual configuration by the administrator is required to select these schemes or key sizes.** • **For HTTPS (administrative Web UI), the TSF generates ECDSA P-256 keys.** • **For TLS key establishment (ECDHE), the TSF supports ECDH using P-256, P-384, and P-521 curves**.

Verdict:

PASS.

5.1.2.2 FCS_CKM.2 CRYPTOGRAPHIC KEY ESTABLISHMENT

5.1.2.2.1 FCS_CKM.2 TSS

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. The intent of this activity is to be able to identify the scheme being used by each service. This would mean, for example, one way to document scheme usage could be as shown in the table below. The information provided in this example does not necessarily have to be included as a table but can be presented in other ways as long as the necessary data is available.

Evaluator Findings:
The evaluator ensured that the supported key establishment schemes correspond to the key generation schemes identified in FCS_CKM.1.1. ST does not specify more than one scheme. The relevant information is found in the following section(s): TOE Summary Specification **FCS_CKM.2 CRYPTOGRAPHIC KEY ESTABLISHMENT (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF performs SP 800-56Ar3 compliant elliptic curve-based key establishment using curves P-256, P-384, and P-521 as part of the TLS handshake, which is as per the key generation schemes identified in FCS_CKM.1.1.** **The relevant NIST CAVP certificates are listed in Table 14.**

Verdict:

PASS.

5.1.2.2.2 FCS_CKM.2 AGD

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

Evaluator Findings:

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

The relevant information is found in the following section(s): **USE OF CRYPTOGRAPHIC KEY GENERATION SCHEMES**

Upon investigation, the evaluator found that the AGD states that: **No manual configuration by the administrator is required to select these schemes or key sizes.**

- **For HTTPS (administrative Web UI), the TSF generates ECDSA P-256 keys.**
- **For TLS key establishment (ECDHE), the TSF supports ECDH using P-256, P-384, and P-521 curves**.

Verdict:

PASS.

5.1.2.3 FCS_CKM.4 CRYPTOGRAPHIC KEY DESTRUCTION

5.1.2.3.1 FCS_CKM.4 TSS

The evaluator shall examine 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 shall confirm that the description of keys and storage locations is consistent with the functions carried out by the TOE (e.g. that all keys for the TOE-specific secure channels and protocols, or that support FPT_APW.EXT.1 and FPT_SKP_EXT.1, are accounted for). In particular, if a TOE claims not to store plaintext keys in non-volatile memory then the evaluator shall check that this is consistent with the operation of the TOE.

Evaluator Findings:
The evaluator examined 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. The relevant information is found in the following section(s): TOE Summary Specification **FCS_CKM.4 CRYPTOGRAPHIC KEY DESTRUCTION (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF destroys keys in RAM by performing an overwrite with zeroes.** **All the keys are in plain-text format. The TSF destroys keys stored in non-volatile memory by logically addressing the storage location and performing an overwrite with zeros. Once the overwrite is complete, the file storing the key is deleted.** **Please see Table 15 for an identification of cryptographic keys, storage locations, generation methods, and timing of zeroization. The TSF destroys TLS private keys stored in non-volatile memory with help of zeroize command.**

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

Evaluator Findings:

The evaluator confirmed that the description of keys and storage locations is consistent with the functions carried out by the TOE (e.g. that all keys for the TOE-specific secure channels and protocols, or that support FPT_APW.EXT.1 and FPT_SKP_EXT.1, are accounted for).

The relevant information is found in the following section(s): TOE Summary Specification **FCS_CKM.4 CRYPTOGRAPHIC KEY DESTRUCTION (CPP_ND_V3.0E)**

Upon investigation, the evaluator found that the TSS states that: **All the keys are in plain-text format. The TSF destroys keys stored in non-volatile memory by logically addressing the storage location and performing an overwrite with zeros. Once the overwrite is complete, the file storing the key is deleted. Please see Table 15 for an identification of cryptographic keys, storage locations, generation methods, and timing of zeroization.**

Zeroization Table

Key	Type	Origin	Storage/Protection	Zeroization
EC Diffie-Hellman Key	Private ECDH P-256, P-384, or P-521	TOE generated	RAM	Keys are overwritten with zeros when session closes
TLS Private Key	Private ECDSA P-256	TOE generated	Restricted Filesystem access	Zeroize command
TLS Encryption Key	128-bit or 256-bit AES	TOE generated	RAM	Keys are overwritten with zeros when session closes
TLS Integrity Key	HMAC-SHA-1, HMAC-SHA-256, or HMAC-SHA-384	TOE generated	RAM	Keys are overwritten with zeros when session closes
User Passwords	SHA-512	User generated	Restricted Filesystem access	Zeroize command

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

Evaluator Findings:

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

The relevant information is found in the following section(s): TOE Summary Specification **FCS_CKM.4 CRYPTOGRAPHIC KEY DESTRUCTION (CPP_ND_V3.0E)**

Upon investigation, the evaluator found that the TSS states that: **The TSF destroys keys in RAM by performing an overwrite with zeroes. All the keys are in plain-text format. The TSF destroys keys stored in non-volatile memory by logically addressing the storage location and performing an overwrite with zeros. Once the overwrite is complete, the file storing the key is deleted. Please see** Error! Reference source not found. **for an identification of cryptographic keys, storage locations, generation methods, and timing of zeroization.**

Zeroization Table

Key	Type	Origin	Storage/Protection	Zeroization
EC Diffie-Hellman Key	Private ECDH P-256, P-384, or P-521	TOE generated	RAM	Keys are overwritten with zeros when session closes
TLS Private Key	Private ECDSA P-256	TOE generated	Restricted Filesystem access	Zeroize command
TLS Encryption Key	128-bit or 256-bit AES	TOE generated	RAM	Keys are overwritten with zeros when session closes
TLS Integrity Key	HMAC-SHA-1, HMAC-SHA-256, or HMAC-SHA-384	TOE generated	RAM	Keys are overwritten with zeros when session closes
User Passwords	SHA-512	User generated	Restricted Filesystem access	Zeroize command

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:

NA. All the keys are stored in plain-text format.

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

Evaluator Findings:
The evaluator checked that the TSS identifies any configurations or circumstances that may not conform to the key destruction requirement (see further discussion in the Guidance Documentation section below). The relevant information is found in the following section(s): TOE Summary Specification **FCS_CKM.4 CRYPTOGRAPHIC KEY DESTRUCTION (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The only situation where the key destruction may be prevented would be if the TOE suffers a loss of power. This situation only impacts on the keys that are stored on the disk. Since the TOE is inaccessible in this situation, administrative zeroization cannot be performed. However, the keys on the disk are protected with restricted file system access.**

Where the ST specifies the use of “a value that does not contain any CSP” to overwrite keys, the evaluator shall examine 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:
NA. The ST does not specify the use of “a value that does not contain any CSP” to overwrite keys; hence, this activity is not applicable.

Verdict:

PASS.

5.1.2.3.2 FCS_CKM.4 AGD

A TOE may be subject to situations that could prevent or delay key destruction in some cases. The evaluator shall check that the guidance documentation identifies configurations or circumstances that may not strictly conform to the key destruction requirement, and that this description is consistent with the relevant parts of the TSS (and any other supporting information used). The evaluator shall check that the guidance documentation provides guidance on situations where key destruction may be delayed at the physical layer. For example, when the TOE does not have full access to the physical memory, it is possible that the storage may be implementing wear-levelling and garbage collection. This may result in additional copies of the key that are logically inaccessible but persist physically. Where available, the TOE might then describe use of the TRIM command[3] and garbage collection to destroy these persistent copies upon their deletion (this would be explained in TSS and Operational Guidance).

Evaluator Findings:
The evaluator checked that the guidance documentation 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 checked that the guidance documentation provides guidance on situations where key destruction may be delayed at the physical layer. The relevant information is found in the following section(s): **Zeroization Process** Upon investigation, the evaluator found that the AGD states that**: The TOE performs key destruction in accordance with the specifications in the Security Target. Keys stored in RAM are destroyed by overwriting them with zeroes.** **Keys stored in non-volatile memory are destroyed by logically addressing the storage location, overwriting it with zeroes, and then deleting the file.** **In the event of a power loss, key destruction may not complete for keys stored on disk. However, such keys are protected by restricted file system access.** • **The TSF zeroizes the EC Diffie-Hellman session keys (P-256, P-384, or P-521) stored in RAM by overwriting them with zeros immediately after the TLS session is terminated.** • **The TSF zeroizes the TLS Encryption Keys (AES-128 or AES-256) stored in RAM by overwriting them with zeros when the corresponding TLS session is closed.** • **The TSF zeroizes the TLS Integrity Keys (HMAC-SHA-1, HMAC-SHA-256, or HMAC-SHA-384) stored in RAM by overwriting them with zeros after session termination.** • **The TSF zeroizes the TLS Private Host Key (ECDSA P-256) stored in non-volatile memory by logically addressing the file system location and performing an overwrite with zeros. Once the overwrite is complete, the key file is deleted.** • **The TSF zeroizes the User Passwords stored in non-volatile memory by performing an overwrite with zeros. Once the overwrite is complete, the key file is deleted** **The only situation where the key destruction may be prevented would be if the TOE suffers a loss of power. This situation only impacts on the keys that are stored on the disk. Since the TOE is inaccessible in this situation, administrative zeroization cannot be performed. However, the keys on the disk are protected with restricted file system access.**

Verdict:

PASS.

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

5.1.2.4.1 FCS_COP.1/DATAENCRYPTION TSS

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

Evaluator Findings:
The evaluator examined the TSS to ensure it identifies the key size(s) and mode(s) supported by the TOE for data encryption/decryption. The relevant information is found in the following section(s): TOE Summary Specification **FCS_COP.1/DATAENCRYPTION CRYPTOGRAPHIC OPERATIONS (AES DATA ENCRYPTION/DECRYPTION) (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF performs AES encryption and decryption in CBC and GCM modes with 128 and 256-bit keys for TLS. The TSF performs AES encryption and decryption in CTR mode with 256-bit keys for Random Bit Generation.**

Verdict:

PASS.

5.1.2.4.2 FCS_COP.1/DATAENCRYPTION AGD

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

Evaluator Findings:
The evaluator verified that the AGD guidance instructs the administrator how to configure the TOE to use the selected mode(s) and key size(s) defined in the Security Target supported by the TOE for data encryption/decryption. The relevant information is found in the following section(s): **Default Configurations** Upon investigation, the evaluator found that the AGD states that: **The following behaviors are enforced automatically without any additional configuration changes:** **•The TOE supports AES encryption and decryption in CBC and GCM modes using 128-bit and 256-bit keys during TLS communications and in CTR mode using 256-bit key during RNG functionality.**

Verdict:

PASS.

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

5.1.2.5.1 FCS_COP.1/SIGGEN TSS

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

Evaluator Findings:
The evaluator examined the TSS to determine that it specifies the cryptographic algorithm and key size supported by the TOE for signature services. The relevant information is found in the following section(s): TOE Summary Specification **FCS_COP.1/SIGGEN CRYPTOGRAPHIC OPERATION (SIGNATURE GENERATION AND VERIFICATION) (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF performs RSA 2048 signature verification as per "FIPS PUB 186-5, "Digital Signature Standard (DSS)", Section 5.4, using PKCS #1 v2.2 Signature Schemes RSASSA-PSS and/or RSASSA-PKCS1v1_5" to verify the integrity and authenticity of updates.** **The TSF performs ECDSA P-256, P-384, P-521 signature generation and verification as per "FIPS PUB 186-5, "Digital Signature Standard (DSS)", Section 6 and NIST SP 800-186 Section 3.2.1, Implementing Weierstrass curves" as part of TLS.**

Verdict:

PASS.

5.1.2.5.2 FCS_COP.1/SIGGEN AGD

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

Evaluator Findings:
The evaluator verified that the AGD guidance instructs the administrator how to configure the TOE to use the selected cryptographic algorithm and key size defined in the Security Target supported by the TOE for signature services. The relevant information is found in the following section(s): **Default Configuration** Upon investigation, the evaluator found that the AGD states that: **The following behaviours are enforced automatically without any additional configuration changes:** • **The TOE supports digital signature generation and verification using:** ○ **RSA Digital Signature Algorithm:** **Supported schemes:** ▪ **RSASSA-PSS (as specified in FIPS PUB 186-4, Section 5.5 and PKCS #1 v2.1)** ▪ **RSASSA-PKCS1v1_5 (as specified in FIPS PUB 186-4, Section 5.5 and PKCS #1 v2.1)** ▪ **ISO/IEC 9796-2 Digital Signature Scheme 2 and Scheme 3** **Supported key size:** ▪ **2048-bit modulus** ○ **Elliptic Curve Digital Signature Algorithm (ECDSA):** **Supported schemes:** ▪ **ECDSA with NIST curves P-256, P-384, and P-521 (as specified in FIPS PUB 186-4, Section 6 and Appendix D)** ▪ **ISO/IEC 14888-3, Section 6.4-compliant ECDSA** **Supported key size:** ▪ **256-bit curve (P-256)**

Verdict:

PASS.

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

5.1.2.6.1 FCS_COP.1/HASH CRYPTOGRAPHIC OPERATION TSS

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

Evaluator Findings:
The evaluator checked that the association of the hash function with other TSF cryptographic functions (for example, the digital signature verification function) is documented in the TSS. The relevant information is found in the following section(s): TOE Summary Specification **FCS_COP.1/HASH CRYPTOGRAPHIC OPERATIONS (HASH ALGORITHM) (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF performs SHA-1, SHA-256, SHA-384, and SHA-512 hashing.** **Hashing is used for the following security functions:** • **NTP – SHA-1** • **Digital Signature generation and verification – SHA-256** • **File Integrity Checking – SHA-256** • **Password Hashing – SHA-512** • **TLS hash function– SHA-1, SHA-256, SHA-384** • **HMAC primitive – SHA-1, SHA-256, SHA-384**

Verdict:

PASS.

5.1.2.6.2 FCS_COP.1/HASH CRYPTOGRAPHIC OPERATION AGD

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

Evaluator Findings:

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

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

Upon investigation, the evaluator found that the AGD states that: **The TOE supports cryptographic hashing services in accordance with a specified cryptographic algorithm SHA-1, SHA-256, SHA-384, SHA-512 and message digest sizes 160, 256, 384, 512 bits that meet the following: ISO/IEC 10118-3:2004. The TOE supports SHA-1, SHA-256, SHA-384, and SHA-512 hashing functions used in NTP, Digital Signature generation and verification, File integrity checking, Password hashing, and HMAC operations. The table below shows the hash algorithms used in the TOE and their specific usages:**

Hash	Usage
SHA-1	**Used for NTP message authentication, HMAC primitive, TLS hash function**
SHA-256	**Used for Digital Signature generation and verification, File integrity checking, HMAC primitive, TLS hash function**
SHA-384	**Supported for hashing functions, HMAC primitive, TLS hash function**
SHA-512	**Used for Password hashing**

The TOE comes preconfigured for these sizes and no additional configuration is required.

Verdict:

PASS.

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

5.1.2.7.1 FCS_COP.1/KEYEDHASH TSS

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

Evaluator Findings:
The evaluator examined the TSS to ensure that it specifies the following values used by the HMAC function: key length, hash function used, block size, and output MAC length used. The relevant information is found in the following section(s): TOE Summary Specification **FCS_COP.1/KEYEDHASH CRYPTOGRAPHIC OPERATION (KEYED HASH ALGORITHM) (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF uses the following HMAC algorithms in TLS:**

Algorithm	Hash Function	Block Size	Key Size	Digest Size
HMAC-SHA-1	SHA-1	512 bits	160 bits	160 bits
HMAC-SHA-256	SHA-256	512 bits	256 bits	256 bits
HMAC-SHA-384	SHA-384	1024 bits	384 bits	384 bits

Verdict:

PASS.

5.1.2.7.2 FCS_COP.1/KEYEDHASH AGD

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

Evaluator Findings:

The evaluator verified that the AGD guidance documents instructs the administrator how to configure the TOE to use the values used by the HMAC function: key length, hash function used, block size, and output MAC length used defined in the Security Target supported by the TOE for keyed hash function.
The relevant information is found in the following section(s): **KEYED HASH CRYPTOGRAPHIC OPERATION (KEYED HASH ALGORITHM)**
Upon investigation, the evaluator found that the AGD states that**: The TOE supports keyed-hash message authentication in accordance with the HMAC algorithm using HMAC-SHA-1, HMAC-SHA-256, and HMAC-SHA-384. The cryptographic key sizes are 160 bits, 256 bits, and 384 bits respectively, with corresponding output MAC lengths of 160, 256, and 384 bits, consistent with ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2."**
• **The associated block sizes are as follows:**
  ○ **HMAC-SHA-1: 512-bit block size**
  ○ **HMAC-SHA-256: 512-bit block size**
  ○ **HMAC-SHA-384: 1024-bit block size**
The TOE comes preconfigured for these algorithms, key sizes, block sizes, and MAC lengths. No additional configuration is required by the administrator.

Verdict:

PASS.

5.1.2.8 FCS_RBG_EXT.1 CRYPTOGRAPHIC OPERATION (RANDOM BIT GENERATION)

5.1.2.8.1 FCS_RBG_EXT.1 TSS

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

Evaluator Findings:
The evaluator examined the TSS and determined that it specifies the DRBG type, identifies the entropy source(s) seeding the DRBG, and state the assumed or calculated min-entropy supplied either separately by each source or the min- entropy contained in the combined seed value. The relevant information is found in the following section(s): TOE Summary Specification **FCS_RBG_EXT.1 RANDOM BIT GENERATION (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF implements an SP 800-90A CTR_DRBG using AES-256. The DRBG is seeded with at least 256-bits of entropy from a platform-based Kernel CPU Time Jitter RNG Entropy Source (JENT version 3.6.0).ESV Certificate E295** **The relevant NIST CAVP certificate is listed in** Error! Reference source not found.**.**

Verdict:

PASS.

5.1.2.8.2 FCS_RBG_EXT.1 AGD

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

Evaluator Findings:
The evaluator confirmed that the guidance documentation contains appropriate instructions for configuring the RNG functionality. The relevant information is found in the following section(s): **Default Configuration and Random Bit Generation** Upon investigation, the evaluator found that the AGD states that: **The following behaviours are enforced automatically without any additional configuration changes:** **The TOE supports AES encryption and decryption, in CBC and GCM modes using 128-bit and 256-bit keys during TLS communications and in CTR mode using 256-bit key during RNG functionality.** **The TOE supports random bit generation services in accordance with ISO/IEC 18031:2011 using CTR_DRBG (AES-256). The TOE comes preconfigured for these sizes and no additional configuration is required.**

Verdict:

PASS.

5.1.3 IDENTIFICATION AND AUTHENTICATION (FIA)

5.1.3.1 FIA_UIA_EXT.1 USER IDENTIFICATION AND AUTHENTICATION

5.1.3.1.1 FIA_UIA_EXT.1 TSS

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

Evaluator Findings:
The evaluator examined the TSS to determine that it describes the logon process for remote authentication mechanism (e.g. SSH public key, Web GUI password, etc.) and optional local authentication mechanisms supported by the TOE. This description shall contain information pertaining to the credentials allowed/used, any protocol transactions that take place, and what constitutes a "successful logon". The relevant information is found in the following section(s): TOE Summary Specification **FIA_UIA_EXT.1 USER IDENTIFICATION AND AUTHENTICATION (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **Remote authentication is based on username/password for the web interface and via external authentication server. The TOE does not expose any interface, through any access method prior to successful login.**

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

Evaluator Findings:
The evaluator examined the TSS and determined that it describes which actions are allowed before administrator identification and authentication. The description shall cover authentication and identification for local and remote TOE administration. The relevant information is found in the following section(s): TOE Summary Specification **FIA_UIA_EXT.1 USER IDENTIFICATION AND AUTHENTICATION (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **Prior to authentication, the TSF only allows users to display the warning banner and automatically generate keys on a new system.**

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

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

For distributed TOEs, the evaluator shall examine the TSS to determine that it describes for each TOE component which actions are allowed before administrator identification and authentication. The description shall cover authentication and identification for remote TOE administration and optionally for local TOE administration if claimed by the ST author. For each TOE component that does not support

authentication of Security Administrators according to FIA_UIA_EXT.1 the TSS shall describe any unauthenticated services/services that are supported by the component.

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

Verdict:

PASS.

5.1.3.1.2 FIA_UIA_EXT.1 AGD

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

Evaluator Findings:

The evaluator examined the guidance documentation and determined that any necessary preparatory steps (e.g., establishing credential material such as pre- shared keys, tunnels, certificates, etc.) to logging in are described. For each supported the login method, the evaluator 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.

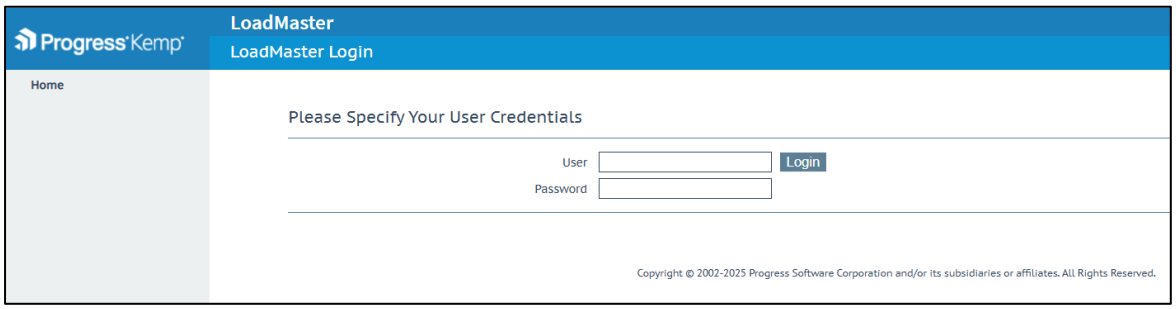
The relevant information is found in the following section(s): **Login via WUI**

Upon investigation, the evaluator found that the AGD states that: **Prior to authentication, the TSF only permits users to access a warning banner (if configured) and to generate system keys automatically during first-time setup. The TOE does not expose any other interface or administrative access prior to successful login. All authenticated access requires a valid username and password.**

The following steps describe the login process via the WUI:

1. Launch a web browser from a laptop that is connected to the device.

2. Log in to the WUI via HTTPS using the IP address assigned during installation, the 'bal' administrative login, and the password you specified during installation.



a. Download the LoadMaster issuing CA ECDSA certificate and install it in the management workstation certificate store and/or the browser certificate store.

3. After initially logging in to the LoadMaster, if Session Management is enabled - some login information is displayed:

a. The last login time and IP address of the current user

b. The number of successful logins by the current user in the last 30 days

c. The total number of failed logins attempts by any user (including unknown usernames) since the last successful login

d. Audit Logs are generated for each action which is performed by a user; either using the API or the WUI.

4. The primary user (bal) always has full permissions. Secondary users may be restricted to certain functions.

5. Only the bal user is permitted to set the Basic Authentication password.

6. No administrative or security-relevant functions are available until the user has successfully authenticated, and the feedback during authentication is obscured.

7. Only authenticated users with the appropriate permissions can view or modify TSF data. Unauthorized users cannot perform any security-related actions.

8. Users must be authenticated before logging on to the LoadMaster.

9. Users with the 'All Permissions' permission set can view the Enable Session Management, Require Basic Authentication, and the Basic Authentication Password fields. However, users with the 'All Permissions' permission set can configure the Failed Login Attempts and Idle Session Timeout values.

10. Users with the 'User Administration' permissions set can view the screen but all buttons and input fields are greyed out.

11. All other users cannot view the WUI Session Management, Currently Active Users or Currently Blocked Users sections of the WUI Configuration screen.

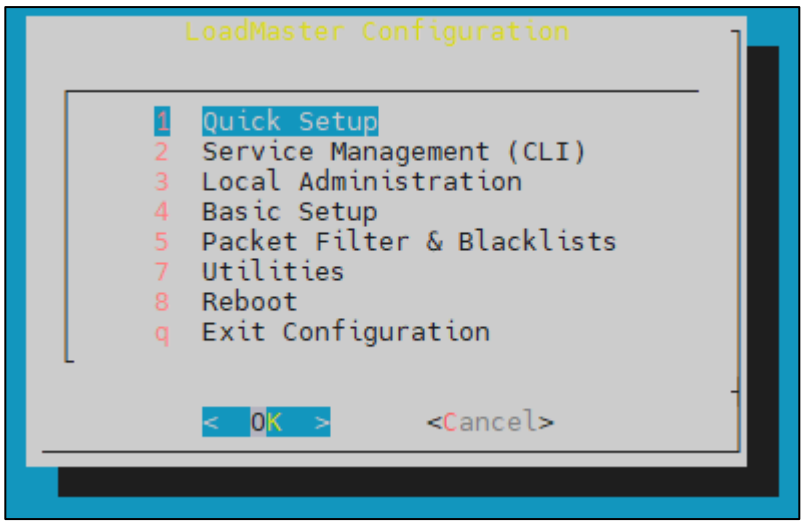
Note: Both the 'bal' and administrator accounts are assigned to the Security Administrator role, which grants permission to manage all TSF functions and configuration. All administrative access to the system must be authenticated through these accounts.

The relevant information is found in the following section(s): **Login via Console**

Upon investigation, the evaluator found that the AGD states that: **The Command Line Interface allows users to interface with the LoadMaster via a command line shell or a menu-based series of options.**

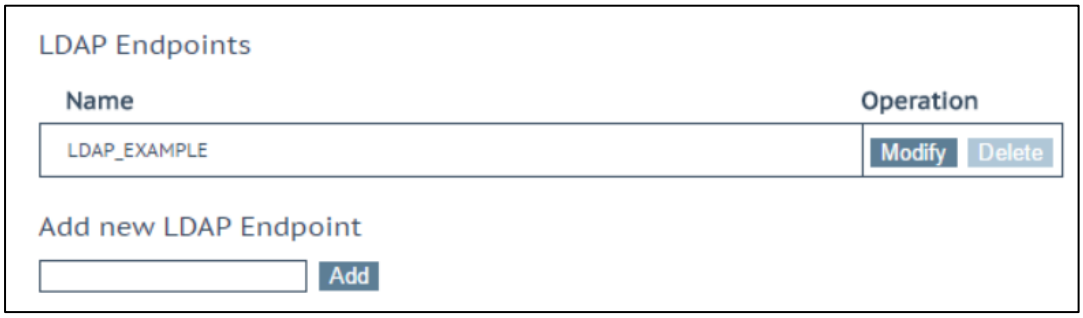
1. **You will need a PC to connect via COM+ (Console) port with a terminal emulation application, or a standard VGA and keyboard. Use a null modem cable (reversal) to connect the COM+ port to the LoadMaster COM port on the rear of the unit.**
2. **The COM+ settings should be 115200, 8, N,1.**
3. **After initially deploying and powering on a LoadMaster, ideally the IP address of the LoadMaster will be obtained via DHCP.**
4. **If the IP address has not been obtained via DHCP, or if the address details of the LoadMaster need to be changed, the console can be used to configure the IP address of the LoadMaster, the default gateway address and the name server addresses.**
5. **To go through this menu, simply log in to the console using the default username and password (bal and 1fourall).**

After successful login, the console displays the LoadMaster configuration menu as shown below:



The relevant information is found in the following section(s): **Setup Admin WUI via LDAP**

Upon investigation, the evaluator found that the AGD states that: **To set up an LDAP domain, click Certificates & Security > LDAP Configuration.**

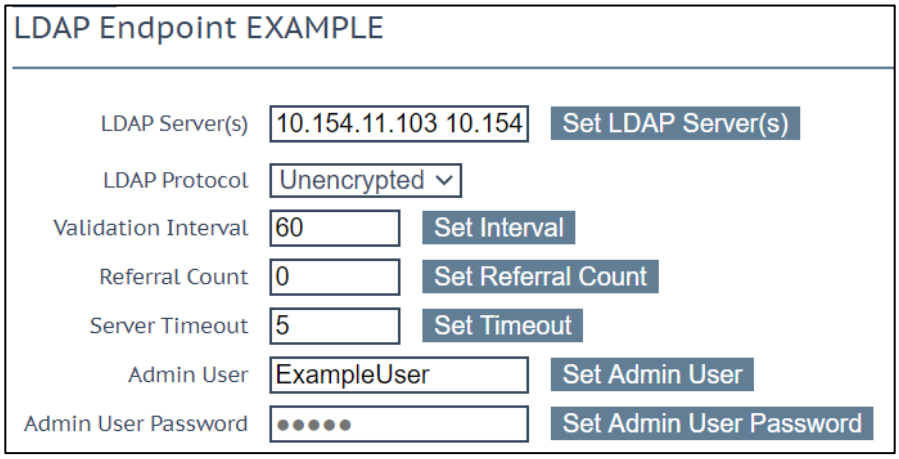


This screen provides a management interface for LDAP endpoints. These LDAP endpoints may be used in three different areas:

- **Health checks**
- **SSO domains**
- **WUI authentication**

Any existing LDAP Endpoints are listed here, with an option to Modify and Delete. If an LDAP endpoint is in use, it cannot be deleted.

There is also an option to add a new LDAP endpoint. Type a name for the endpoint and click Add. Spaces and special characters are not permitted in the LDAP endpoint name.



- **LDAP Server(s)**

  Specify a space-separated list of LDAP servers to be used. Port numbers can also be specified if required. If you have multiple domains and are using Permitted Groups, sometimes it is necessary to include the Global Catalog port number, otherwise the Permitted Groups will fail. The default port is 3628. For example, 10.110.20.23:3268.

  The LoadMaster uses OCSP to check the validity of the server certificates supplied by configured LDAPS servers. If these checks fail, connections to the server are not permitted.

- **LDAP Protocol**

  Select the transport protocol to use when communicating with the LDAP server.

  If you create an SSO domain with the Authentication Protocol set to Certificates, ensure to set the LDAP Protocol to LDAPS in the LDAP endpoint.

- **Validation Interval**

  Specify how often you should revalidate the user with the LDAP server.

- **Referral Count**

  The LoadMaster offers beta functionality to support LDAP referral replies from Active Directory Domain Controllers. If this is set to 0, referral support is not enabled. Set this field to a value between 1 and 10 to enable referral chasing. The number specified will limit the number of hops (referrals chased).

- **Server Timeout**

  Specify the LDAP server timeout in seconds. The default value is 5. Valid values range from 5 to 60.

- **Admin User**

  Type the username of an administrator user.

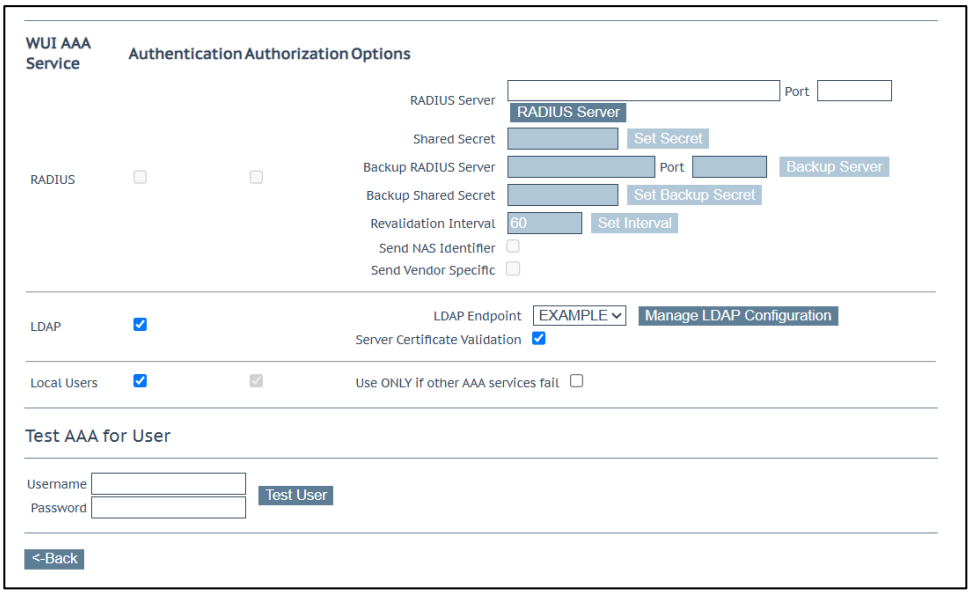
- **Admin User Password**

  Type the password for the specified administrator user.

To set up admin WUI (bal account) access via LDAP/AD:

a. **In the left frame menu, click Certificates & Security > Remote Access.**

b. **In the page at right, click the WUI Authorization Option button.**



c. **Select the LDAP option.**

d. **Select the LDAP Endpoint that is created.**

e. **Add Remote User Groups if created.**

f. **Add the Domain and enable Server Certificate Validation option.**

Note: LDAPS channel is restricted to TLSv1.2

Verdict:

PASS.

5.1.4 SECURITY MANAGEMENT (FMT)

5.1.4.1 FMT_MOF.1/MANUALUPDATE MANAGEMENT OF SECURITY FUNCTIONS BEHAVIOUR

5.1.4.1.1 FMT_MOF.1/MANUALUPDATE TSS

For distributed TOEs see Section 5.1.4.1. There are no specific requirements for non-distributed TOEs.

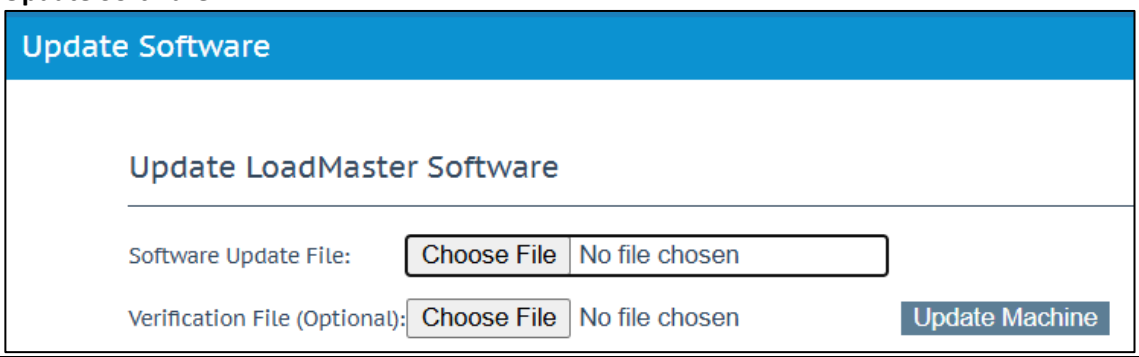
5.1.4.1.2 FMT_MOF.1/MANUALUPDATE AGD

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

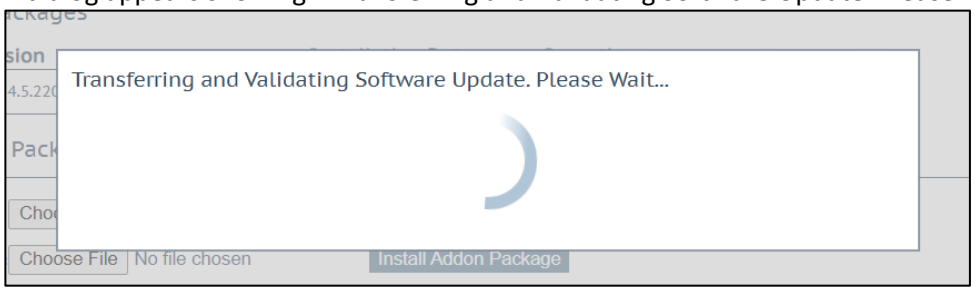
Evaluator Findings:

The evaluator examined the guidance documentation and determined that any necessary steps to perform manual update are described. The guidance documentation also provides warnings regarding functions that may cease to operate during the update (if applicable).
The relevant information is found in the following section(s): **INSTALLING AN UPDATE IMAGE**
Upon investigation, the evaluator found that the AGD states that: To install an update image on LoadMaster:

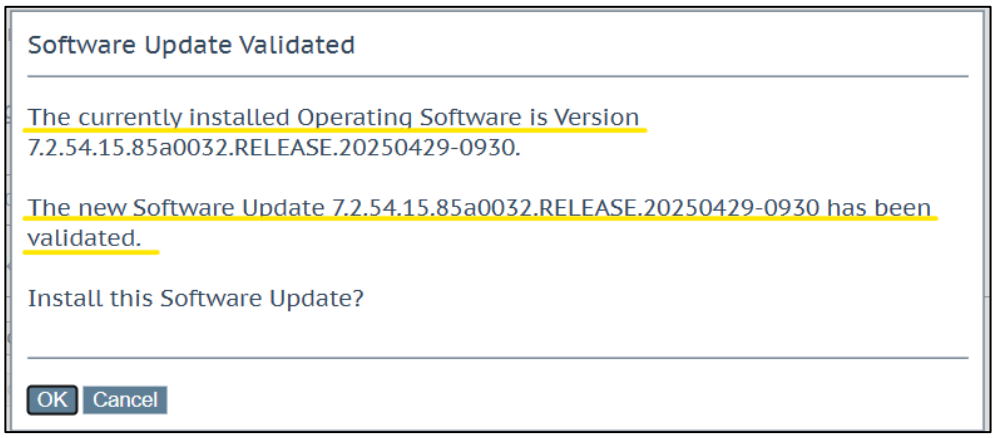
- Unzip the archive received from Progress on a laptop or other device with a browser.
  - The archive contains the update image and an XML file.
- Open the LoadMaster WUI and navigate to **System Configuration > System Administration > Update Software.**



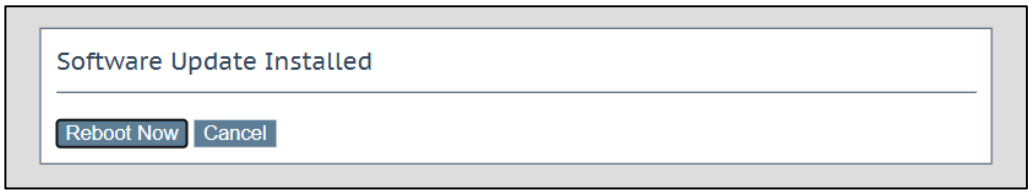
- Click on the Browse button next to Software Update File and select the update image from Step 1.
- Click on the Browse button next to Verification File and select the XML file from Step 1.
- Click Update Machine.
- A dialog appears showing: “Transferring and Validating Software Update. Please Wait...”



- Once validation is successful, a confirmation window is displayed showing both the currently installed version and the validated new version.



- Click OK to proceed or Cancel to abort the update process.
- On After confirming the update, the system installs the software and then prompts the administrator with Reboot Now or Cancel.

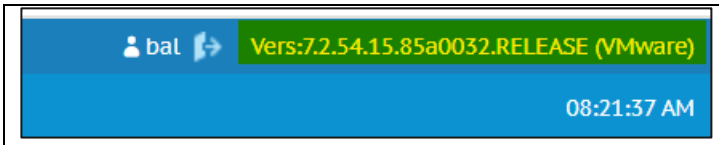


  - Clicking Reboot Now reboots the TOE and activates the new version.
  - Clicking Cancel aborts the update process. The TOE does not retain the uploaded version, and the system continues running the previously active version.

Note: The TOE does not support delayed activation. If the reboot is not performed immediately after installation, the update is discarded and must be re-uploaded and installed again.

- The currently active version of the TOE is displayed in the top-right corner of the Home page screen.

The TSF allows the Security Administrator to query the currently running version of the software and initiate software updates via HTTPS WUI. The currently active software version is displayed on the Home page of the TOE, specifically in the top-right corner. To view this, the administrator must log in to the WUI using valid credentials. Upon successful authentication, the Home page is displayed by default, showing the active version information.



The WUI will display a warning message indicating that the device is rebooting, and all functions will cease to operate during the reboot. Once the reboot is complete and the WUI becomes accessible again, the administrator can log in to the TOE.

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

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

Verdict:

PASS.

5.1.4.2 FMT_MTD.1/COREDATA MANAGEMENT OF TSF DATA

5.1.4.2.1 FMT_MTD.1/COREDATA TSS

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

Evaluator Findings:
The evaluator confirmed that the TSS details how the ability to manipulate the TSF data through these interfaces is disallowed for non-administrative users. The relevant information is found in the following section(s): TOE Summary Specification **FMT_MTD.1/COREDATA MANAGEMENT OF TSF DATA (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF displays a warning banner prior to user authentication. There are no administrative functions available for unauthorized users. All administrators must be authenticated and authorized to perform any activity that can alter TSF data. The TSF restricts the ability to manage TSF data to security administrators.**

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:
If the TOE supports handling of X.509v3 certificates and implements a trust store, the evaluator examined the TSS and determined that it contains sufficient information to describe how the ability to manage the TOE's trust store is restricted. The relevant information is found in the following section(s): TOE Summary Specification **FMT_MTD.1/COREDATA MANAGEMENT OF TSF DATA (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TOE provides a trust store to handle the X.509v3 certificates. The TOE restrict access to the trust store with help of permissions so that only Security Administrator users can import or delete certificates from the trust store.**

Verdict:

PASS.

5.1.4.2.2 FMT_MTD.1/COREDATA AGD

The evaluator shall review the guidance documentation to determine that each of the TSF-data-manipulating functions implemented in response to the requirements of the 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 entire AGD 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 following sections in the AGD describes the configuration available for each of the TSF-data manipulating functions available on the TOE, which is consistent with the requirements of the cPP identified in the ST. **'User Creation'** **'Set Minimum Password Length'** **'Intermediate Certificate'** **'Generate CSR (Certificate Signing Request)'** **'Logging out'** **'Log in via WUI'** **'Setting Date/Time'** **'Setup Admin UI Access via LDAP'** **'Secure Remote Logging'** **'Idle Session Timeout'** **'Failed Login Attempts'** **'Set the WUI Banner'** **'Set the CLI Banner'** **'Configuring NTP server'** **'Disable SSH Access'**

'Installing an Update Image'

'Login VIA console'

The section entitled "CC Configuration Process" covers each of the TSF-data manipulating functions available on the TOE and explains that when administrators log in with role-based credentials, their access is limited to commands they have privileges and permissions to use based on the Common Criteria standards. No management functionality is available prior to successful identification and authentication of the users. Network management communication paths are protected against modification and disclosure by TLS. The set of subsections within the section entitled "Intermediate Certificate" and "Generate CSR (Certificate Signing Request)" describe administrative actions that administrators can perform to manage certificates and generate certificate-signing-request respectively.

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

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

Evaluator Findings:

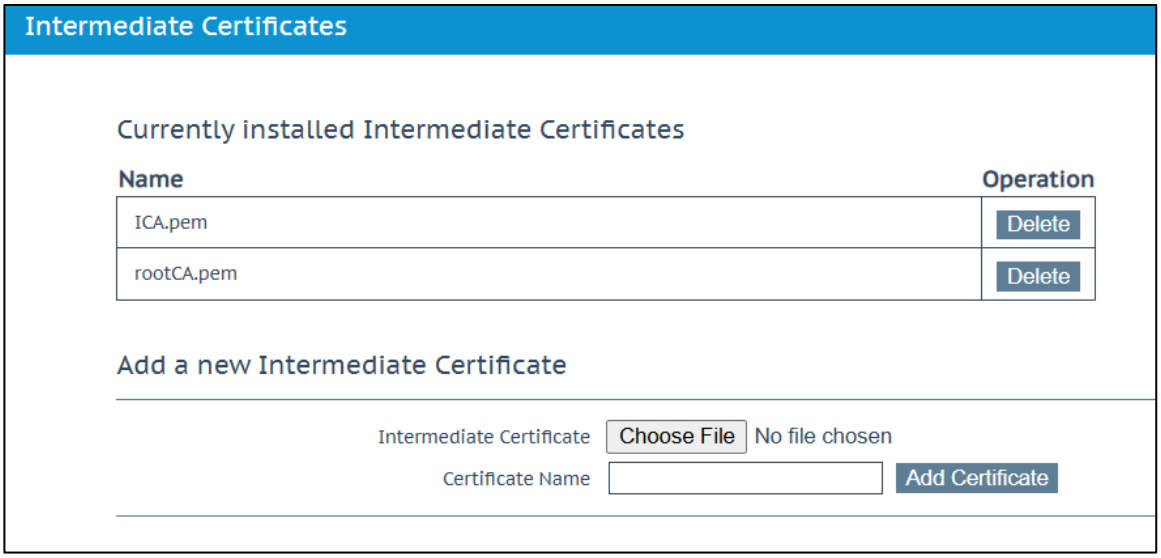
If the TOE supports handling of X.509v3 certificates and provides a trust store, the evaluator reviewed the guidance documentation section tilted **LOG IN via WUI** and determined that it provides sufficient information for the administrator to configure and maintain the trust store in a secure way. If the TOE supports loading of CA certificates, the evaluator reviewed the guidance documentation section titled **Intermediate Certificate** and determined that it provides sufficient information for the administrator to securely load CA certificates into the trust store. The evaluator also reviewed the guidance documentation section titled **ENABLING AND CONFIGURING TLS CIPHER SUITES FOR SECURE COMMUNICATION** and determined that it explains how to designate a CA certificate a trust anchor. The relevant information is found in the following section(s): **LOG IN via WUI**

Upon investigation, the evaluator found that the AGD states that: **The following steps describe the login process via the WUI:**

1. **Launch a web browser from a laptop that is connected to the device.**
2. **Log in to the WUI via HTTPS using the IP address assigned during installation, the 'bal' administrative login, and the password you specified during installation.**
    a. **Download the LoadMaster issuing CA ECDSA certificate and install it in the management workstation certificate store and/or the browser certificate store.**

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

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



This screen shows a list of the installed certificates and the name assigned to them. The Intermediate Certificates field allows you to assign intermediate and root certificates.

If you already have a certificate, or you have received one from a CSR, you can install the certificate by clicking the Choose File button. Navigate to and select the certificate and then enter the desired Certificate Name. The name can only contain alpha characters with a maximum of 32 characters.

Uploading several consecutive intermediate certificates within a single piece of text, as practiced by some certificate vendors such as GoDaddy, is allowed. The uploaded file is split into the individual certificates.

You can also delete the certificates by clicking the 'Delete' button shown in the first screenshot of this section.

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

The relevant information is found in the following section(s): **ENABLING AND CONFIGURING TLS CIPHER SUITES FOR SECURE COMMUNICATION**

Upon investigation, the evaluator found that the AGD states that: **To configure the TLS cipher suites used by the TOE for secure WUI access, perform the following steps:**

1. **Upload Root and Intermediate CA Certificates**

   Before enabling certificate-based admin WUI access:

   - **Navigate to Certificates & Security → Intermediate Certificates.**
   - **Upload the issuing CA and Root CA certificates required to validate administrator certificates for client authentication.**

Verdict:

PASS.

5.1.4.3 FMT_SMF.1 SPECIFICATION OF MANAGEMENT FUNCTIONS

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

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

Evaluator Findings:
The evaluator examined the TSS section 6, the Guidance Documentation and the TOE as observed during all other testing and confirmed that the management functions specified in FMT_SMF.1 are provided by the TOE. The relevant information is found in the following section(s): TOE Summary Specification **FMT_SMF.1 SPECIFICATION OF MANAGEMENT FUNCTIONS (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF supports local (Console) and remote (WUI) administrative interfaces.** **The following management functions are available at the Console:** • **Ability to start and stop services.** • **Ability to manage the cryptographic keys.** • **Ability to configure the cryptographic functionality.** • **Ability to administer the TOE locally.** • **Ability to configure the local session inactivity time before session termination or locking.** • **Ability to set the time which is used for time-stamps.** • **Ability to update the TOE, and to verify the updates using digital signature capability prior to installing those updates.** **The following management functions are available via the WUI:** • **Ability to administer the TOE remotely.** • **Ability to manage the cryptographic keys.** • **Ability to configure the cryptographic functionality.** • **Ability to configure the access banner.** • **Ability to configure the remote session inactivity time before session termination.** • **Ability to configure the local session inactivity time before session termination or locking.** • **Ability to update the TOE, and to verify the updates using digital signature capability prior to installing those updates.** • **Ability to configure the list of TOE-provided services available before an entity is identified and authenticated, as specified in FIA_UIA_EXT.1.**

- **Ability to configure the list of supported (D)TLS ciphers.**
- **Ability to re-enable an Administrator account.**
- **Ability to set the time which is used for time-stamps.**
- **Ability to configure NTP.**
- **Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors.**
- **Ability to modify the behaviour of the transmission of audit data to an external IT entity.**
- **Ability to generate Certificate Signing Request (CSR) and process CA certificate response.**
- **Ability to configure the authentication failure parameters for FIA_AFL.1.**

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

Evaluator Findings:

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

The relevant information is found in the following section(s): TOE Summary Specification **FMT_SMF.1 SPECIFICATION OF MANAGEMENT FUNCTIONS (CPP_ND_V3.0E)**

Upon investigation, the evaluator found that the TSS states that **The TSF supports local (Console) and remote (WUI) administrative interfaces.**

The following management functions are available at the Console:

- **Ability to start and stop services.**
- **Ability to manage the cryptographic keys.**
- **Ability to configure the cryptographic functionality.**
- **Ability to administer the TOE locally.**
- **Ability to configure the local session inactivity time before session termination or locking.**
- **Ability to set the time which is used for time-stamps.**
- **Ability to update the TOE, and to verify the updates using digital signature capability prior to installing those updates.**

The following management functions are available via the WUI:

- **Ability to administer the TOE remotely.**
- **Ability to manage the cryptographic keys.**
- **Ability to configure the cryptographic functionality.**
- **Ability to configure the access banner.**
- **Ability to configure the remote session inactivity time before session termination.**
- **Ability to configure the local session inactivity time before session termination or locking.**
- **Ability to update the TOE, and to verify the updates using digital signature capability prior to installing those updates.**
- **Ability to configure the list of TOE-provided services available before an entity is identified and authenticated, as specified in FIA_UIA_EXT.1.**
- **Ability to configure the list of supported (D)TLS ciphers.**
- **Ability to re-enable an Administrator account.**
- **Ability to set the time which is used for time-stamps.**
- **Ability to configure NTP.**
- **Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors.**
- **Ability to modify the behaviour of the transmission of audit data to an external IT entity.**
- **Ability to generate Certificate Signing Request (CSR) and process CA certificate response.**
- **Ability to configure the authentication failure parameters for FIA_AFL.1.**

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

Evaluator Findings:
The evaluator examined the TSS and the Guidance Documentation to verify they both describe the local administrative interface. The relevant information is found in the following section(s): TOE Summary Specification **FMT_SMF.1 SPECIFICATION OF MANAGEMENT FUNCTIONS (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF supports local (Console) and remote (WUI) administrative interfaces.** The relevant information is found in the following section(s): **Login via Console** Upon investigation, the evaluator found that the AGD states that: **The Command Line Interface allows users to interface with the LoadMaster via a command line shell or a menu-based series of options.** **1. You will need a PC to connect via COM+ (Console) port with a terminal emulation application, or a standard VGA and keyboard. Use a null modem cable (reversal) to connect the COM+ port to the LoadMaster COM port on the rear of the unit.** **2. The COM+ settings should be 115200, 8, N,1.** **3. After initially deploying and powering on a LoadMaster, ideally the IP address of the LoadMaster will be obtained via DHCP.** **4. If the IP address has not been obtained via DHCP, or if the address details of the LoadMaster need to be changed, the console can be used to configure the IP address of the LoadMaster, the default gateway address and the name server addresses.** **5. To go through this menu, simply log in to the console using the default username and password (bal and 1fourall).**

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

Evaluator Findings:
The evaluator ensured the Guidance Documentation includes appropriate warnings for the administrator to ensure the interface is local. The relevant information is found in the following section(s): **SET THE CLI BANNER** Upon investigation, the evaluator found that the AGD states that**: 1. In the left frame menu in WUI, click Certificates & Security > Remote Access.** **2. Check the Allow Remote SSH Access box, type in an SSH Pre-Auth Banner and click Set Pre-Auth Message. This banner is also used for the CLI (even if SSH is disabled).**

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

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

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

Evaluator Findings:
The evaluator checked that the TOE behaviour observed during testing of the configured SFRs is as described in the TSS **FMT_SMF.1** and the Guidance Documentation **Section 6.1, Section 6.2, Section 6.5, Section 6.8, Section 6.9, Section 6.10, Section 6.11, Section 6.12, Section 6.13, Section 6.14, Section 6.15, Section 6.17, Section 6.20, Section 6.21, Section 6.22, Section 6.23, Section 6.24, Section 6.25, Section 6.28, Section 6.32 ,Section 6.33 , Section 6.37, Section 6.38 and Section 6.39.**

(If configure local audit is selected) The evaluator shall examine the TSS and Guidance Documentation to ensure that a description of the logging implementation is described in enough detail to determine how log files are maintained on the TOE.

Evaluator Findings:
NA. Configure local audit is not selected in ST.

Verdict:

PASS.

5.1.4.3.2 FMT_SMF.1 AGD

See Section 5.1.4.3.1.

5.1.4.4 FMT_SMR.2 RESTRICTIONS ON SECURITY ROLES

5.1.4.4.1 FMT_SMR.2 TSS

The evaluator shall examine the TSS to determine that it details the TOE supported roles and any restrictions of the roles involving administration of the TOE (e.g. if local administrators and remote administrators have different privileges or if several types of administrators with different privileges are supported by the TOE).

Evaluator Findings:
The evaluator examined the TSS and determined that it details the TOE supported roles and any restrictions of the roles involving administration of the TOE. The relevant information is found in the following section(s): TOE Summary Specification **FMT_SMR.2 RESTRICTIONS ON SECURITY ROLES (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF supports a 'bal' account (superuser) and administrator accounts. Both account types belong to the Security Administrator role. The factory default username is bal. The factory default user retains the highest level of access. All users created have a subset of the access permitted by the default account. Changing roles for users takes effect in real-time. Roles can be combined and are mutually exclusive.** **The various permission roles are described below.** **Real Servers** This role permits the following operations on Real Servers:

- Add
- Modify
- Delete
- Enable
- Disable

Virtual Services
This role relates to managing Virtual Services. This includes SubVSs. Virtual Service actions permitted vary depending on whether or not the Allow Extended Permissions option is enabled. For further information, refer to the Virtual Service Permissions section.
Rules
This role permits managing content rules. Rule actions permitted include adding, deleting and modifying.
System Backup
This role permits performing system backups.
Certificate Creation
This role permits managing SSL certificates. Certificate management includes adding, deleting and modifying SSL certificates.
Intermediate Certificates
This role permits managing intermediate certificates. This includes adding and deleting intermediate certificates.
Certificate Backup
This role permits the ability to export and import certificates.
User Administration
This role is allowed access to all functionality within the System Configuration > System Administration > User Management screen, for all user management.
GEO Control
This role provides the ability to manage GEO settings, if relevant. For further information on GEO, refer to the GEO, Feature Description on the Documentation Page.
Add Virtual Services
This role is only visible if the Allow Extended Permissions check box is enabled. This role relates to managing Virtual Services. This includes SubVSs. Refer to the Virtual Service Permissions section for further details on the permissions provided by this option.
All Permissions
This role provides all permissions, except the ability to change the bal password.
Virtual Service Permissions

There are two permissions relating to Virtual Services - Virtual Services and Add Virtual Services.

Verdict:

PASS.

5.1.4.4.2 FMT_SMR.2 AGD

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

Evaluator Findings:

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

The relevant information is found in the following section(s): **Login via WUI**

Upon investigation, the evaluator found that the AGD states that: **The following steps describe the login process via the WUI:**

1. **Launch a web browser from a laptop that is connected to the device.**
2. **Log in to the WUI via HTTPS using the IP address assigned during installation, the 'bal' administrative login, and the password you specified during installation.**
    a. **Download the LoadMaster issuing CA ECDSA certificate and install it in the management workstation certificate store and/or the browser certificate store.**
3. **After initially logging in to the LoadMaster, if Session Management is enabled - some login information is displayed:**
    a. **The last login time and IP address of the current user**
    b. **The number of successful logins by the current user in the last 30 days**
    c. **The total number of failed logins attempts by any user (including unknown usernames) since the last successful login**
    d. **Audit Logs are generated for each action which is performed by a user; either using the API or the WUI.**
4. **The primary user (bal) always has full permissions. Secondary users may be restricted to certain functions.**
5. **Only the bal user is permitted to set the Basic Authentication password.**
6. **No administrative or security-relevant functions are available until the user has successfully authenticated, and the feedback during authentication is obscured.**
7. **Only authenticated users with the appropriate permissions can view or modify TSF data. Unauthorized users cannot perform any security-related actions.**
8. **Users must be authenticated before logging on to the LoadMaster**

The relevant information is found in the following section(s): **Login via Console**

Upon investigation, the evaluator found that the AGD states that: **The Command Line Interface allows users to interface with the LoadMaster via a command line shell or a menu-based series of options.**

1. **You will need a PC to connect via COM+ (Console) port with a terminal emulation application, or a standard VGA and keyboard. Use a null modem cable (reversal) to connect the COM+ port to the LoadMaster COM port on the rear of the unit.**
2. **The COM+ settings should be 115200, 8, N,1.**
3. **After initially deploying and powering on a LoadMaster, ideally the IP address of the LoadMaster will be obtained via DHCP.**

4. **If the IP address has not been obtained via DHCP, or if the address details of the LoadMaster need to be changed, the console can be used to configure the IP address of the LoadMaster, the default gateway address and the name server addresses.**
5. **To go through this menu, simply log in to the console using the default username and password (bal and 1fourall).**

The relevant information is found in the following section(s): **AUTHENTICATION AND ROLE-BASED ACCESS CONTROL**

Upon investigation, the evaluator found that the AGD states that: **No administrative or security-relevant functions are available until the user has successfully authenticated, and the feedback during authentication is obscured. Only authenticated users with the appropriate permissions can view or modify TSF data. Unauthorized users cannot perform any security-related actions.**

Users must be authenticated before logging on to the LoadMaster.

Users with the 'All Permissions' permission set can view the Enable Session Management, Require Basic Authentication, and the Basic Authentication Password fields. However, users with the 'All Permissions' permission set can configure the Failed Login Attempts and Idle Session Timeout values.

Users with the 'User Administration' permissions set can view the screen but all buttons and input fields are greyed out.

All other users cannot view the WUI Session Management, Currently Active Users or Currently Blocked Users sections of the WUI Configuration screen.

Note: Both the 'bal' and administrator accounts are assigned to the Security Administrator role, which grants permission to manage all TSF functions and configuration. All administrative access to the system must be authenticated through these accounts.

Verdict:

PASS.

5.1.5 PROTECTION OF THE TSF (FPT)

5.1.5.1 FPT_SKP_EXT.1 PROTECTION OF TSF DATA (FOR READING OF ALL SYMMETRIC KEYS)

5.1.5.1.1 FPT_SKP_EXT.1 TSS

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, by any enabled role as outlined in the application note. If these values are not stored in plaintext, the TSS shall describe how they are protected/obscured.

Evaluator Findings:
The evaluator examined the TSS and determined that it details how any pre- shared keys, symmetric keys, and private keys are stored and that they are unable to be viewed through an interface designed specifically for that purpose, as outlined in the application note. If these values are not stored in plaintext, the TSS describes how they are protected/obscured. The relevant information is found in the following section(s): TOE Summary Specification **FPT_SKP_EXT.1 PROTECTION OF TSF DATA (FOR READING OF ALL SYMMETRIC KEYS) (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF prevents reading symmetric and private keys. The private TLS certificate keys are protected through UI restrictions that prevent the security administrators from reading the keys. All other symmetric and private keys are only held in RAM and can only be accessed by the processes performing TLS. The TSF does not utilize pre-shared keys.** The relevant information is found in the following section(s): **FCS_CKM.4 CRYPTOGRAPHIC KEY DESTRUCTION (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **All the keys are in plain-text format.**

Verdict:

PASS.

5.1.5.1.2 FPT_SKP_EXT.1 AGD

None.

5.1.5.2 FPT_STM_EXT.1 RELIABLE TIME STAMPS

5.1.5.2.1 FPT_STM_EXT.1 TSS

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

Evaluator Findings:
The evaluator examined the TSS and ensured that it lists each security function that makes use of time, and that it provides a description of how the time is maintained and considered reliable in the context of each of the time related functions. The relevant information is found in the following section(s): TOE Summary Specification **FPT_STM_EXT.1 RELIABLE TIME STAMPS (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF maintains the date and time using the clock provided by the underlying hardware. This date and time is used as the time stamp that is applied to TOE generated audit records and used to track inactivity of administrative sessions. Also, this date and time is used for x509 Certificate expiration validation.** **The time can be manually updated by a Security Administrator or automatically updated using NTP synchronization.**

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

Evaluator Findings:
NA. “obtain time from the underlying virtualization system” is not selected in the ST.

Verdict:

PASS.

5.1.5.2.2 FPT_STM_EXT.1 AGD

The evaluator shall examine 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.

Evaluator Findings:

The evaluator examined the guidance documentation and ensured that 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.

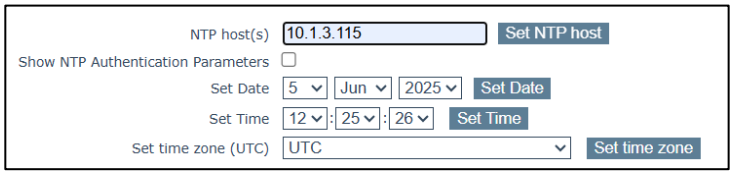
The relevant information is found in the following section(s): **SETTING DATE/TIME**

Upon investigation, the evaluator found that the AGD states that: **The current system date and time, maintained using the underlying hardware clock, is displayed. This timestamp is applied to all audit records and is used to track the inactivity of administrative sessions.**

The date and time of LoadMaster can be configured in two ways:

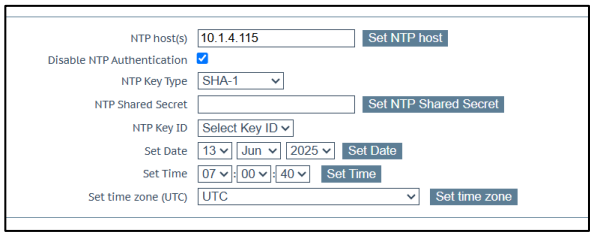
- **Automatic Synchronization via NTP. for further details and configuration refer section 6.10.**
- **Manual Configuration: A Security Administrator can directly modify the date and time fields to set the correct system time.**

1. **In the left frame menu, click System Configuration > System Administration > Date/Time:**
   - **Set Date: Set the date on the LoadMaster.**
   - **Set Time: Set the time on the LoadMaster.**
   - **Set time zone: Set the time zone where the LoadMaster is located.**



The relevant information is found in the following section(s): Configuring NTP Server

Upon investigation, the evaluator found that the AGD states that: You can manually configure the date and time of LoadMaster or leverage an NTP server. It supports time updates using NTPv4 only. The LoadMaster authentications updates using an administrator configured symmetric key and SHA-1. By default, the TOE rejects broadcast and multicast time updates. The TOE allows up to 10 NTP time sources to be configured.



In the left frame menu, click System Configuration > System Administration > Date/Time:

a. **NTP host(s): Specify the host which is to be used as the NTP server. Multiple NTP servers can be configured with this setting.**

b. **Show NTP Authentication Parameters: Enable the Show NTP Authentication Parameters check box to display the parameters that are needed to support NTP**

authenticated requests. Disable the Show NTP Authentication Parameters checkbox to hide.

c. **NTP Key Type: Select SHA-1 from the drop-down menu for the NTP key type to ensure the integrity and authenticity of the timestamp data exchanged between the TOE and the NTP server.**

d. **NTP Shared Secret: The NTP shared secret string. The NTP secret can be a maximum of 20 ASCII characters long or 40 hexadecimal characters long.**

e. **NTP Key ID: Select the NTP key ID. The values range from 1 to 99. Different key IDs can be used for different servers.**

f. **Set Date: Set the date on the LoadMaster.**

g. **Set Time: Set the time on the LoadMaster.**

h. **Set time zone: Set the time zone where the LoadMaster is located.**

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:
N/A. The TOE does not support obtaining time from the underlying VS.

Verdict:

PASS.

5.1.5.3 FPT_TST_EXT.1 TSF TESTING

5.1.5.3.1 FPT_TST_EXT.1 TSS **[TD0836]**

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

Evaluator Findings:
The evaluator examined the TSS and ensured that it details each of the self-tests that are identified by the SFR; 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" is used). The relevant information is found in the following section(s): TOE Summary Specification **FPT_TST_EXT.1 TSF TESTING (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF performs the following tests at power-up:** • **File systems checks** **During boot up the TSF checks file system by verifying the metadata and that it is mounted correctly.** • **SHA-256 Software integrity checks** **The TSF generates a SHA-256 has of the firmware image and compares it with the stored value.** • **Cryptographic algorithm known answer tests** **For each cryptographic algorithm, the TSF performs a sample cryptographic operation using know values and compares the output with the expected value.** • **Cryptographic algorithm pairwise constancy test** **For each cryptographic algorithm with a key-pair, the TSF performs a sample operation using know value and compares the output with the corresponding key-pair.** • **Health test of the noise source** **This is a continuous health-test that checks the number of occurrences of 6 different bit patterns in each 256-bit output from the noise source. It checks if any of the pattern counts are outside of predetermined thresholds. If more than 128 of the most recent 256 256-bit samples fails, the Entropy Source cease to output data.**

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

Evaluator Findings:
The evaluator ensured that the TSS makes an argument that the tests are sufficient to demonstrate that the TSF is operating correctly. The relevant information is found in the following section(s): TOE Summary Specification **FPT_TST_EXT.1 TSF TESTING (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **These tests collectively ensures the TSF is operating correctly.**

If more than one failure response is listed in FPT_TST_EXT.1.2, the evaluator shall examine the TSS to ensure it clarifies which response is associated with which type of failure.

Evaluator Findings:
If more than one failure response is listed in FPT_TST_EXT.1.2, the evaluator shall examine the TSS to ensure it clarifies which response is associated with which type of failure. The relevant information is found in the following section(s): TOE Summary Specification **FPT_TST_EXT.1 TSF TESTING (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **If any of the tests fail, the TSF disables the affected functionality or halts. The TSF will boot with cryptographic service disabled if the known answer tests, pairwise consistency test, or noise source health test fail**.

For distributed TOEs the evaluator shall examine the TSS to ensure that it details which TOE component performs which self-tests and when these self-tests are run. The evaluator shall also examine the TSS to ensure it describes how the TOE reacts if one or more TOE components fail self-testing (e.g. halting and displaying an error message; failover behaviour).

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

Verdict:

PASS.

5.1.5.3.2 FPT_TST_EXT.1 AGD

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

Evaluator Findings:

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

The relevant information is found in the following section(s): **Start-Up and Self-Test**

Upon investigation, the evaluator found that the AGD states that: **The TSF runs a suite of the following self-tests [during initial start-up (on power on)] to demonstrate the correct operation of the TSF:**

- **File systems checks**

  During boot up the TSF checks file system by verifying the metadata and that it is mounted correctly.

- **SHA-256 Software integrity checks**

  The TSF generates a SHA-256 has of the firmware image and compares it with the stored value.

- **Cryptographic algorithm known answer tests**

  For each cryptographic algorithm, the TSF performs a sample cryptographic operation using know values and compares the output with the expected value.

- **Cryptographic algorithm pairwise constancy test**

  For each cryptographic algorithm with a key-pair, the TSF performs a sample operation using know value and compares the output with the corresponding key-pair.

- **Health test of the noise source**

  This is a continuous health-test that checks the number of occurrences of 6 different bit patterns in each 256-bit output from the noise source. It checks if any of the pattern counts are outside of predetermined thresholds. If more than 128 of the most recent 256 256-bit samples fails, the Entropy Source cease to output data.

Failure Handling:

- **If any of these tests fail:**
  - **The affected functionality (e.g., cryptographic services) is disabled.**
  - **If critical failures occur (e.g., in cryptographic algorithm KATs, consistency tests, or entropy source), the TSF halts boot or proceeds with cryptographic services disabled.**

  These self-tests ensure the TOE is operating securely and reliably, In such cases, the administrator may review the failure condition and proceed with rebooting the TOE.

<u>Verification of Self-Test Failure:</u>

If the self-tests fail during system startup, the following log message is generated:

"Crypto Self-Tests failed"

This indicates that the system has not passed the required cryptographic integrity checks and may not proceed with normal operation.

Verification of Self-Test Success:

Upon successful completion of the self-tests, the following log message is generated:

2026-02-17T13:26:12+00:00 lb100 logger: script Self-Test passed

2026-02-17T13:26:12+00:00 lb100 logger: sha1 Self-Test passed

2026-02-17T13:26:12+00:00 lb100 logger: sha160 Self-Test passed

2026-02-17T13:26:12+00:00 lb100 logger: sha256 Self-Test passed

2026-02-17T13:26:12+00:00 lb100 logger: sha384 Self-Test passed

2026-02-17T13:26:12+00:00 lb100 logger: sha512 Self-Test passed

2026-02-17T13:26:12+00:00 lb100 logger: hmacsha1 Self-Test passed

2026-02-17T13:26:12+00:00 lb100 logger: hmacsha160 Self-Test passed

2026-02-17T13:26:12+00:00 lb100 logger: hmacsha256 Self-Test passed

2026-02-17T13:26:12+00:00 lb100 logger: hmacsha384 Self-Test passed

2026-02-17T13:26:12+00:00 lb100 logger: hmacsha512 Self-Test passed

2026-02-17T13:26:12+00:00 lb100 logger: aescbc128 Self-Test passed

2026-02-17T13:26:12+00:00 lb100 logger: aescbc256 Self-Test passed

2026-02-17T13:26:12+00:00 lb100 logger: aesecb128 Self-Test passed

2026-02-17T13:26:12+00:00 lb100 logger: aesecb256 Self-Test passed

2026-02-17T13:26:12+00:00 lb100 logger: rand Self-Test passed

2026-02-17T13:26:12+00:00 lb100 logger: RSA Sign and Verify Self-Test passed

2026-02-17T13:26:13+00:00 lb100 logger: EC Self-Test passed

2026-02-17T13:26:13+00:00 lb100 logger: Crypto Self-Tests passed

- **Administrators can check the system logs for these messages to confirm that the cryptographic self-tests and system initialization completed successfully during startup.**

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

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

Verdict:

PASS.

5.1.5.4 FPT_TUD_EXT.1 TRUSTED UPDATE

5.1.5.4.1 FPT_TUD_EXT.1 TSS

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

Evaluator Findings:
The evaluator verified that the TSS describes how to query the currently active version. If a trusted update can be installed on the TOE with a delayed activation, the TSS describes how and when the inactive version becomes active. The evaluator verified this description. The relevant information is found in the following section(s): TOE Summary Specification **FPT_TUD_EXT.1 TRUSTED UPDATE (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF allows the Security Administrator to query the currently running version of software via HTTPS WUI. The currently active software version is displayed on the Home page of the TOE, specifically in the top-right corner. To view this, the administrator must log in to the WUI using valid credentials. Upon successful authentication, the Home page is displayed by default, showing the active version information.** **Delayed activation of a trusted update is not supported by the TOE.**

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 of the software before installation and that installation fails if the verification fails.

Evaluator Findings:
The evaluator verified 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 verified that the description includes a digital signature verification of the software before installation and that installation fails if the verification fails. The relevant information is found in the following section(s): TOE Summary Specification **FPT_TUD_EXT.1 TRUSTED UPDATE (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TOE supports manual update mechanism. Updates are available on support website. The Security Administrator obtains updates by downloading them on administrator's workstation. When the Security Administrator uploads a firmware update using HTTPS web GUI, the TSF performs digital signature verification of the update. Prior to installing an update, the TSF verifies an RSA 2048 signature on the update to ensure the update is authentic.** **If the digital signature check is successful, the TSF installs the update. If the digital signature check detects tampering with the update and/or signature, the TSF presents the user with an error message and discards the update.**

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

Evaluator Findings:
The evaluator verified that the TSS describes the method by which the digital signature is verified to include how the candidate updates are obtained, the processing associated with verifying the digital signature of the update, and the actions that take place for both successful and unsuccessful signature verification. The relevant information is found in the following section(s): TOE Summary Specification **FPT_TUD_EXT.1 TRUSTED UPDATE (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **When the Security Administrator uploads a firmware update using HTTPS web GUI, the TSF performs digital signature verification of the update. Prior to installing an update, the TSF verifies an RSA 2048 signature on the update to ensure the update is authentic.** **If the digital signature check is successful, the TSF installs the update. If the digital signature check detects tampering with the update and/or signature, the TSF presents the user with an error message and discards the update.**

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:
NA. The ST does select ‘support automatic checking for updates’ or ‘support automatic updates’, hence this assurance activity is not applicable.

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

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

Verdict:

PASS.

5.1.5.4.2 FPT_TUD_EXT.1 AGD

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

Evaluator Findings:
The evaluator verified that the guidance documentation describes how to query the currently active version. If a trusted update can be installed on the TOE with a delayed activation, the guidance documentation describes how to query the loaded but inactive version. The relevant information is found in the following section(s): **Installing an Update Image** Upon investigation, the evaluator found that the AGD states that: **The TSF allows the Security Administrator to query the currently running version of the software and initiate software updates via HTTPS WUI. The currently active software version is displayed on the Home page of the TOE, specifically in the top-right corner. To view this, the administrator must log in to the WUI using valid credentials. Upon successful authentication, the Home page is displayed by default, showing the active version information.** Additionally, the evaluator found that the AGD activity states that: **Note: The TOE does not support delayed activation. If the reboot is not performed immediately after installation, the update is discarded and must be re-uploaded and installed again.**

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

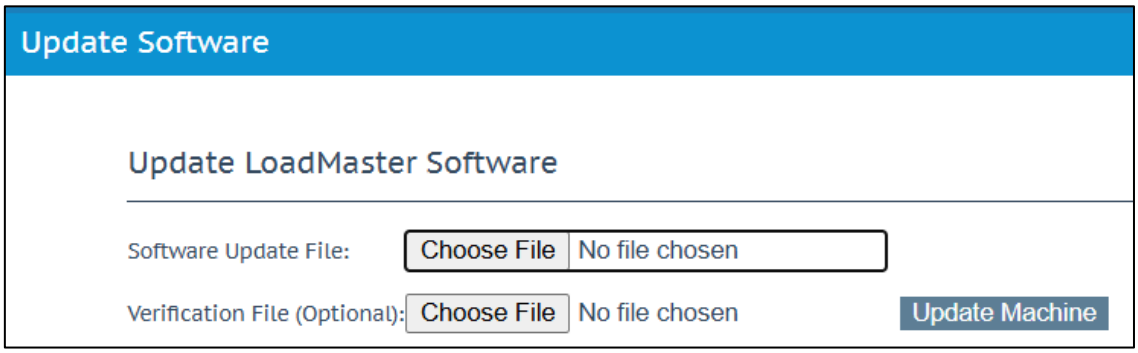
Evaluator Findings:
The evaluator verified that the guidance documentation describes how the verification of the authenticity of the update is performed (digital signature verification). The description includes the procedures for successful and unsuccessful verification. The description corresponds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

The relevant information is found in the following section(s): **Installing an Update Image**

Upon investigation, the evaluator found that the AGD states that: **The software image must be downloaded by the Security Administrator from the official support website to a trusted administrative workstation. The TOE performs a mandatory digital signature verification using RSA-2048 before installing the software update. If the signature is valid, the update proceeds as described. If the digital signature verification fails (due to tampering or an invalid image), the TSF discards the update, displays an error message, and generates an audit log entry. The TOE does not support hash-based verification for update validation.**

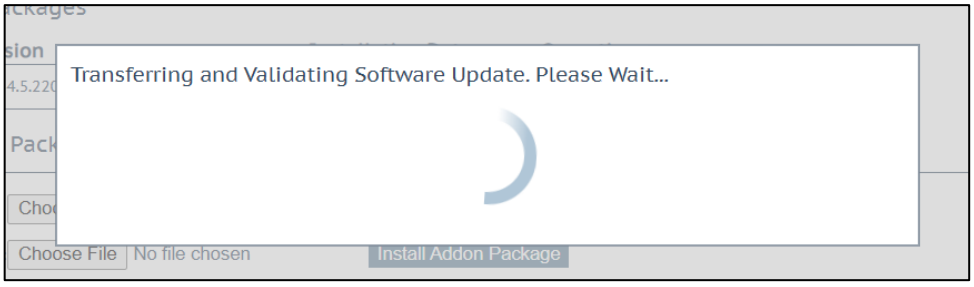
1. **Additionally, the evaluator found that the AGD activity states that: Unzip the archive received from Progress on a laptop or other device with a browser.**
    - **The archive contains the update image and an XML file.**
2. **Open the LoadMaster WUI and navigate to:**

    System Configuration > System Administration > Update Software.

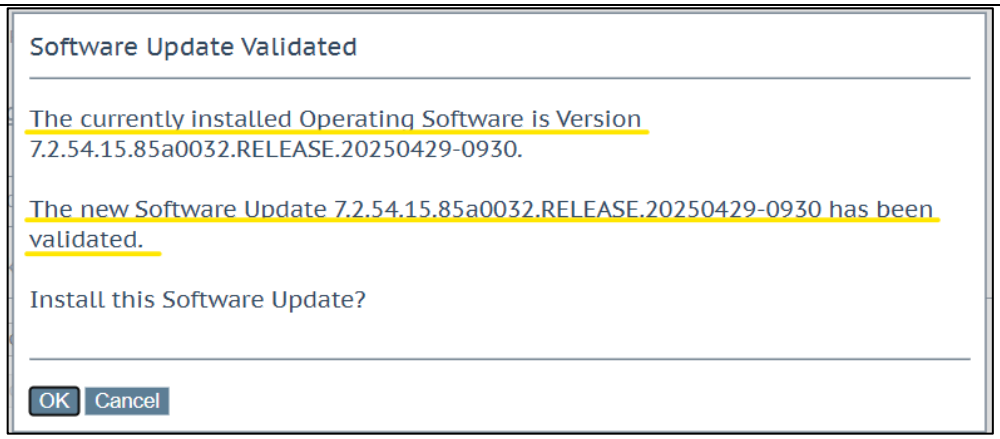


3. **Click on the Browse button next to Software Update File and select the update image from Step 1.**
4. **Click on the Browse button next to Verification File and select the XML file from Step 1.**
5. **Click Update Machine.**

A dialog appears showing: "Transferring and Validating Software Update. Please Wait..."

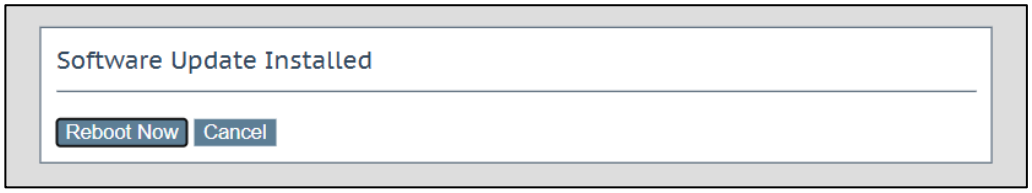


Once validation is successful, a confirmation window is displayed showing both the currently installed version and the validated new version.



6. Click OK to proceed or Cancel to abort the update process.

7. On After confirming the update, the system installs the software and then prompts the administrator with Reboot Now or Cancel.



- **Clicking Reboot Now reboots the TOE and activates the new version.**
- **Clicking Cancel aborts the update process. The TOE does not retain the uploaded version, and the system continues running the previously active version.**

Note: The TOE does not support delayed activation. If the reboot is not performed immediately after installation, the update is discarded and must be re-uploaded and installed again.

8. The currently active version of the TOE is displayed in the top-right corner of the Home page screen.

Additionally, the evaluator found that the AGD activity states that: **If the signature is valid, the update proceeds as described. If the digital signature verification fails (due to tampering or an invalid image), the TSF discards the update, displays an error message, and generates an audit log entry. The TOE does not support hash-based verification for update validation. The TOE rejects updates if an invalid software image or mismatched XML file is provided.**

Note: The TOE does not support hash-based verification for update validation. The TOE rejects updates if an invalid software image or mismatched XML file is provided.

For distributed TOEs the evaluator shall verify that the guidance documentation describes how the versions of individual TOE components are determined for FPT_TUD_EXT.1, how all TOE components are updated, and the error conditions that may arise from checking or applying the update (e.g. failure of signature verification, or exceeding available storage space) along with appropriate recovery actions. The guidance documentation only has to describe the procedures relevant for the Security

Administrator; it does not need to give information about the internal communication that takes place when applying updates.

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

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

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

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

Evaluator Findings:
NA. The evaluator examined the Security Target and verified that a certificate-based mechanism is not used for software update digital signature verification.

Verdict:

PASS.

5.1.6 TOE ACCESS (FTA)

5.1.6.1 FTA_SSL.3 TSF-INITIATED TERMINATION

5.1.6.1.1 FTA_SSL.3 TSS

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

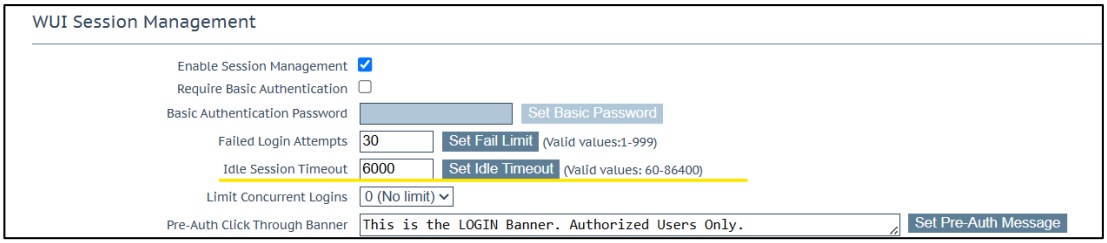
Evaluator Findings:
The evaluator examined the TSS and determined that it details the administrative remote session termination and the related inactivity time period. The relevant information is found in the following section(s): TOE Summary Specification **FTA_SSL.3 TSF-INITIATED TERMINATION (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF terminates remote sessions after 60-86400 seconds of inactivity. If a remote user session is inactive for a configured period of time, the session will be terminated and will require re-identification and authentication to establish a new session. When the user logs back in, the inactivity timer will be activated for the new session.**

Verdict:

PASS.

5.1.6.1.2 FTA_SSL.3 AGD

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

Evaluator Findings:
The evaluator confirmed that the guidance documentation includes instructions for configuring the inactivity time period for remote administrative session termination. The relevant information is found in the following section(s): **Idle Session Timeout** Upon investigation, the evaluator found that the AGD states that: **TOE allows administrators to configure an idle session timeout to automatically log out inactive sessions.** **To configure the idle timeout setting:** **1. In the left frame menu in the WUI, navigate to:** **System Configuration > Certificate & Security > Admin WUI Access > Idle Session Timeout.** **The length of time (in seconds) a user can be idle (no activity recorded) before they are logged out of the session.** **2. Enter the desired timeout value in seconds.** • **Valid values range from 60 seconds (1 minute) to 86400 seconds (24 hours).** • **This setting determines how long a session can remain idle before it is automatically terminated by the TSF.**      **The TSF allows the administrator to terminate their own active local or remote interactive session through the session management options in the WUI.** **The TSF enforces an administrator-configurable session inactivity timeout, which can be set from 1 to 1440 minutes. If an administrative session remains idle beyond the configured timeout, the TSF automatically terminates the session. This configured timeout period applies to both remote WUI sessions and the Local console sessions.**

Verdict:

PASS.

5.1.6.2 FTA_SSL.4 USER-INITIATED TERMINATION

5.1.6.2.1 FTA_SSL.4 TSS

The evaluator shall examine the TSS to determine that it details how the remote administrative session (and if applicable the local administrative session) are terminated.

Evaluator Findings:
The evaluator examined the TSS and determined that it details how the remote administrative session are terminated. The relevant information is found in the following section(s): TOE Summary Specification **FTA_SSL.4 USER-INITIATED TERMINATION (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF allows the administrator to terminate the administrator's own local and remote interactive sessions with the help of command, or the options mentioned in the guidance documentation.** **For WUI:** • After logging in via the WUI, the user can log out by clicking the Logout button, located in the top right-hand corner of the screen. **For CLI:** • When accessing the console, the main menu is displayed by default. • To log out of the console session, user must press q to Quit the session.

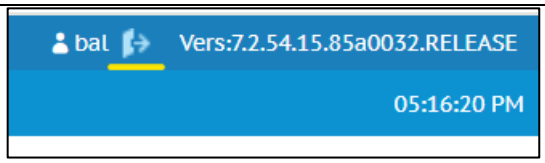
Verdict:

PASS.

5.1.6.2.2 FTA_SSL.4 AGD

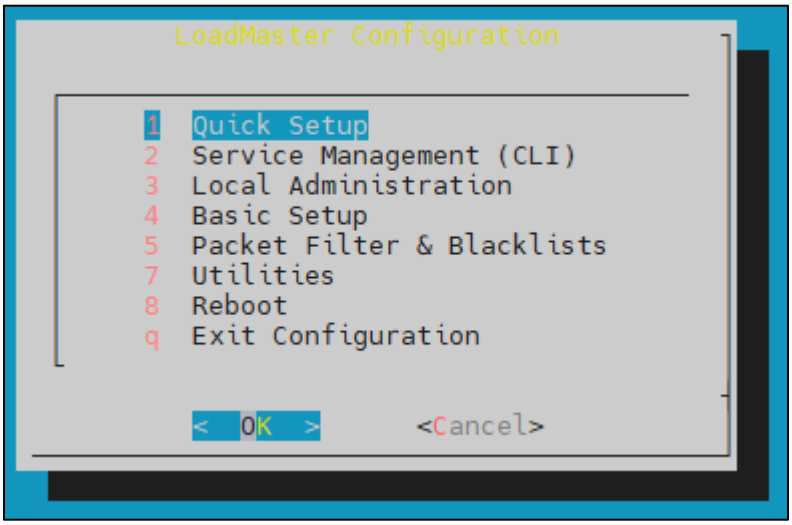
The evaluator shall confirm that the guidance documentation states how to terminate a remote interactive session (and if applicable the local administrative session).

Evaluator Findings:
The evaluator confirmed that the guidance documentation states how to terminate a remote interactive session (and if applicable the local administrative session). The relevant information is found in the following section(s): **Log Out** Upon investigation, the evaluator found that the AGD states that: **For WUI:** • **After logging in via the WUI, the user can log out by clicking the Logout button, located in the top right-hand corner of the screen.**



For CLI:

1. **When accessing the console, the main menu is displayed by default.**



2. **To log out of the console session, press q to Quit the session.**

Verdict:

PASS.

5.1.6.3 FTA_TAB.1 DEFAULT TOE ACCESS BANNERS

5.1.6.3.1 FTA_TAB.1 TSS

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

Evaluator Findings:
The evaluator checked the TSS and ensured that it details each administrative method of access (local and remote) available to the Security Administrator (e.g., serial port, SSH, HTTPS). The relevant information is found in the following section(s): TOE Summary Specification **FTA_TAB.1 DEFAULT TOE ACCESS BANNERS (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF displays a configurable message before establishing a Local (Serial port) and Remote Administrative session (HTTPS). This banner will be displayed prior to allowing Security Administrator access through those interfaces.**

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

Evaluator Findings:
The evaluator checked the TSS and ensured that all administrative methods of access available to the Security Administrator are listed and that the TSS states that the TOE is displaying an advisory notice and a consent warning message for each administrative method of access. The advisory notice and the consent warning message might be different for different administrative methods of access and might be configured during initial configuration (e.g. via configuration file). The relevant information is found in the following section(s): TOE Summary Specification **FTA_TAB.1 DEFAULT TOE ACCESS BANNERS (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF displays a configurable message before establishing a Local (Serial port) and Remote Administrative session (HTTPS). This banner will be displayed prior to allowing Security Administrator access through those interfaces.**

Verdict:

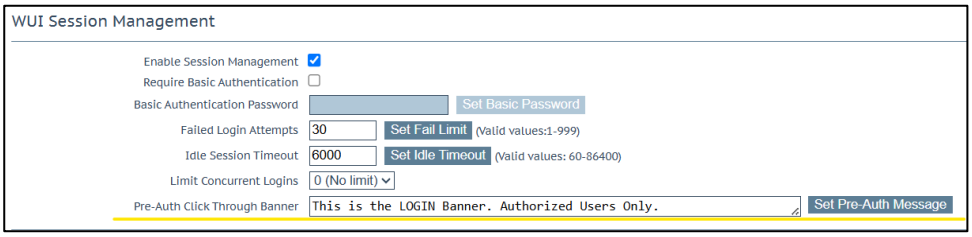
PASS.

5.1.6.3.2 FTA_TAB.1 AGD

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

Evaluator Findings:
The evaluator examined the guidance documentation and ensured that it describes how to configure the banner message. The relevant information is found in the following section(s): **SET THE WUI BANNER** Upon investigation, the evaluator found that the AGD states that: **The TSF displays a configurable pre-authentication banner before establishing Remote (HTTPS/WUI) administrative session.**

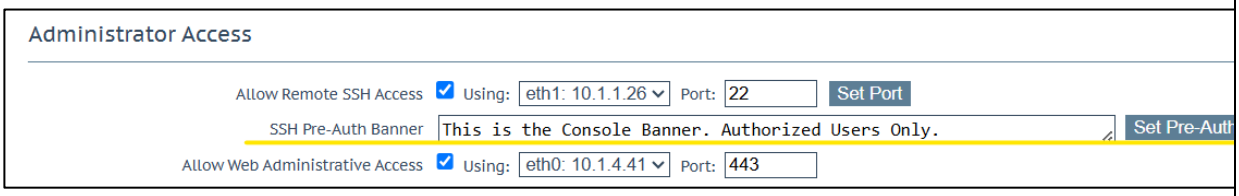
1. **In the left frame menu, click Certificates & Security > Admin WUI Access**
2. **Type in a Pre-Auth Click Through Banner and click Set Pre-Auth Message**



The relevant information is found in the following section(s): **SET THE CLI BANNER**

Upon investigation, the evaluator found that the AGD states that: **The TSF displays a configurable pre-authentication banner before establishing Local (Console/Serial) session.**

1. **In the left frame menu in WUI, click Certificates & Security > Remote Access.**
2. **Check the Allow Remote SSH Access box, type in an SSH Pre-Auth Banner and click Set Pre-Auth Message. This banner is also used for the CLI (even if SSH is disabled).**



Note: To ensure the SSH Pre-Auth Banner applies only to console access and to effectively disable remote SSH access, set the "Using" option to the console interface IP instead of "All Networks," and configure the port as needed. This restricts SSH access to the specified interface, preventing remote access via other network interfaces.

Verdict:

PASS.

5.1.7 TRUSTED PATH/CHANNELS (FTP)

5.1.7.1 FTP_ITC.1 INTER-TSF TRUSTED CHANNEL

5.1.7.1.1 FTP_ITC.1 TSS

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

Evaluator Findings:
The evaluator examined the TSS and determined that, for all communications with authorized IT entities identified in the requirement, each secure communication mechanism is identified in terms of the allowed protocols for that IT entity, whether the TOE acts as a server or a client, and the method of assured identification of the non-TSF endpoint. The relevant information is found in the following section(s): TOE Summary Specification **FTP_ITC.1 INTER-TSF TRUSTED CHANNEL (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF communicates with the syslog server and LDAP server using TLS v1.2 as described in the descriptions of FCS_TLSC_EXT.1 above.**

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

Evaluator Findings:
The evaluator also confirmed that all secure communication mechanisms are described in sufficient detail to allow the evaluator to match them to the cryptographic protocol Security Functional Requirements listed in the ST. The relevant information is found in the following section(s): TOE Summary Specification **FTP_ITC.1 INTER-TSF TRUSTED CHANNEL (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF communicates with the syslog server and LDAP server using TLS v1.2 as described in the descriptions of FCS_TLSC_EXT.1 above. The protocols listed are consistent with those specified in the requirement.**

Verdict:

PASS.

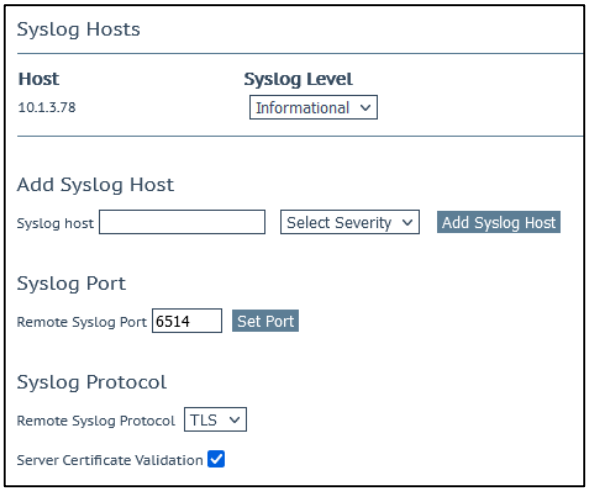
5.1.7.1.2 FTP_ITC.1 AGD

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

Evaluator Findings:
The evaluator confirmed that the AGD contains instructions for establishing the allowed protocols with each authorized IT entity, and that it contains recovery instructions should a connection be unintentionally broken. The relevant information is found in the following section(s): **Secure Remote Logging** Upon investigation, the evaluator found that the AGD states that: The LoadMaster can produce various warning and error messages using the syslog protocol. These messages are normally stored locally. Syslog messages are transmitted securely using TLS to remote servers. The LoadMaster uses OCSP to check the validity of the server certificates supplied by configured syslog servers. If these checks fail, connections to the server are not permitted. **1.** In the left frame menu, click **System Configuration > System Log Files > Syslog Options.**

2. By entering the relevant IP address in the **Syslog host** text box and select the severity and click **Add Syslog Host.**
3. Also enter the **Syslog Port** enter *any port other than 601* and click on **Set Port**.
4. Select the **Syslog Protocol** either TCP, UDP or TLS.
5. Enable the Server Certificate Validation.

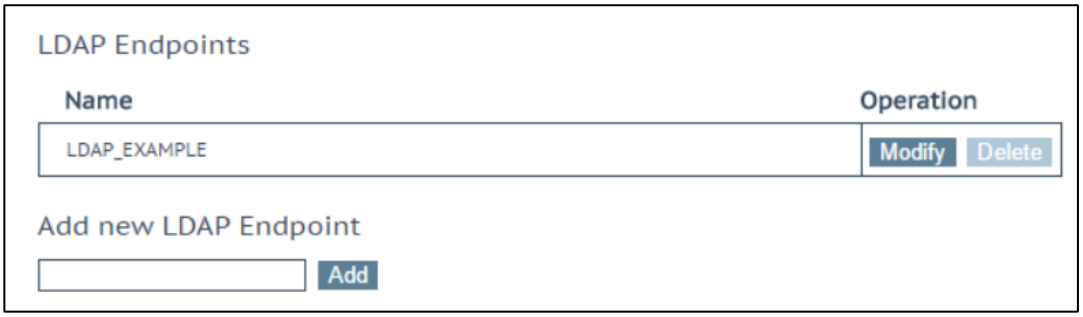
Note: secure syslog channel is restricted to TLSv1.2.



NOTE: If the TLS connection to the syslog server is unintentionally broken, the administrator should verify the connectivity to the syslog server. Once connectivity is restored, the administrator must manually re-initiate the syslog connection.

The relevant information is found in the following section(s): **Setup Admin WUI access via LDAP**

Upon investigation, the evaluator found that the AGD states that: **To set up an LDAP domain, click Certificates & Security > LDAP Configuration.**

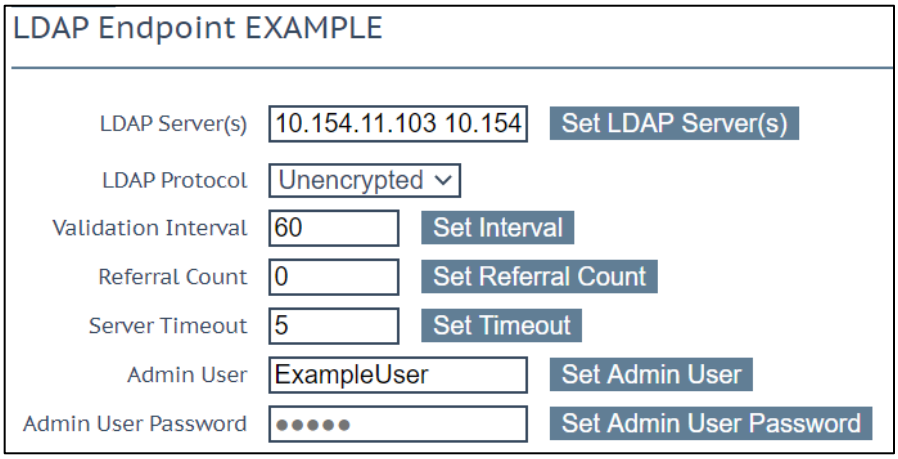


This screen provides a management interface for LDAP endpoints. These LDAP endpoints may be used in three different areas:

- **Health checks**
- **SSO domains**
- **WUI authentication**

Any existing LDAP Endpoints are listed here, with an option to Modify and Delete. If an LDAP endpoint is in use, it cannot be deleted.

There is also an option to add a new LDAP endpoint. Type a name for the endpoint and click Add. Spaces and special characters are not permitted in the LDAP endpoint name.



- **LDAP Server(s)**

  Specify a space-separated list of LDAP servers to be used. Port numbers can also be specified if required. If you have multiple domains and are using Permitted Groups, sometimes it is necessary to include the Global Catalog port number, otherwise the Permitted Groups will fail. The default port is 3628. For example, 10.110.20.23:3268.

  The LoadMaster uses OCSP to check the validity of the server certificates supplied by configured LDAPS servers. If these checks fail, connections to the server are not permitted.

- **LDAP Protocol**

  Select the transport protocol to use when communicating with the LDAP server.

  If you create an SSO domain with the Authentication Protocol set to Certificates, ensure to set the LDAP Protocol to LDAPS in the LDAP endpoint.

- **Validation Interval**

  Specify how often you should revalidate the user with the LDAP server.

- **Referral Count**

  The LoadMaster offers beta functionality to support LDAP referral replies from Active Directory Domain Controllers. If this is set to 0, referral support is not enabled. Set this field to a value between 1 and 10 to enable referral chasing. The number specified will limit the number of hops (referrals chased).

- **Server Timeout**

Specify the LDAP server timeout in seconds. The default value is 5. Valid values range from 5 to 60.

- **Admin User**

  Type the username of an administrator user.

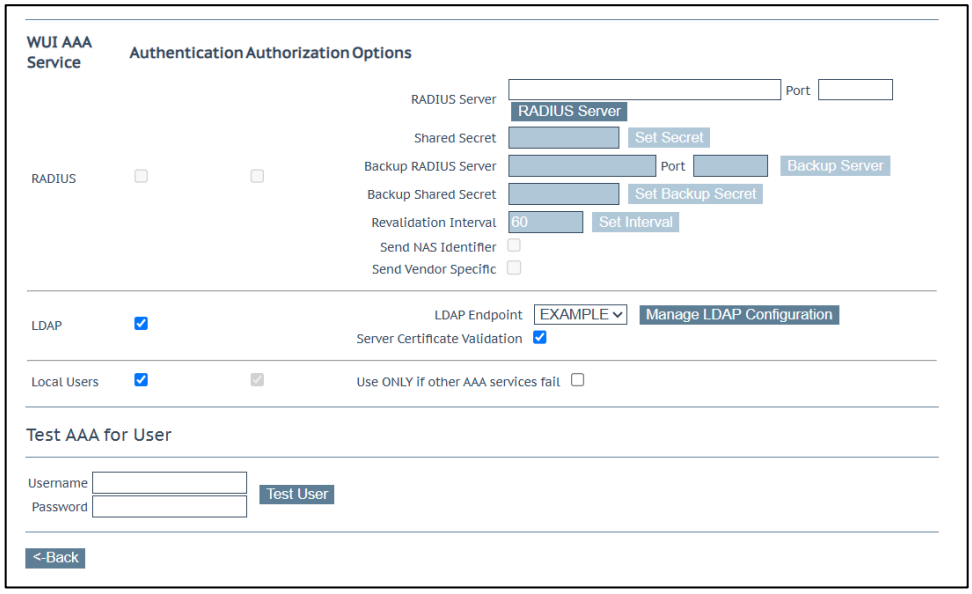
- **Admin User Password**

  Type the password for the specified administrator user.

To set up admin WUI (bal account) access via LDAP/AD:

g. In the left frame menu, click Certificates & Security > Remote Access.

h. In the page at right, click the WUI Authorization Option button.



i. Select the LDAP option.

j. Select the LDAP Endpoint that is created.

k. Add Remote User Groups if created.

l. Add the Domain and enable Server Certificate Validation option.

Note: LDAPS channel is restricted to TLSv1.2

Verdict:

PASS.

5.1.7.2 FTP_TRP.1/ADMIN TRUSTED PATH

5.1.7.2.1 FTP_TRP.1/ADMIN TSS

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

Evaluator Findings:
The evaluator examined the TSS and determined that the methods of remote TOE administration are indicated, along with how those communications are protected. The relevant information is found in the following section(s): TOE Summary Specification **FTP_TRP.1/ADMIN TRUSTED PATH (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF provides a trusted path with remote administrators using TLS/HTTPS as described in FCS_TLSS_EXT.1, and FCS_HTTPS_EXT.1.**

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

Evaluator Findings:
The evaluator also confirmed that all protocols listed in the TSS in support of TOE administration are consistent with those specified in the requirement, and are included in the requirements in the ST. The relevant information is found in the following section(s): TOE Summary Specification **FTP_TRP.1/ADMIN TRUSTED PATH (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The protocols listed are consistent with those specified in the requirement.**

Verdict:

PASS.

5.1.7.2.2 FTP_TRP.1/ADMIN AGD

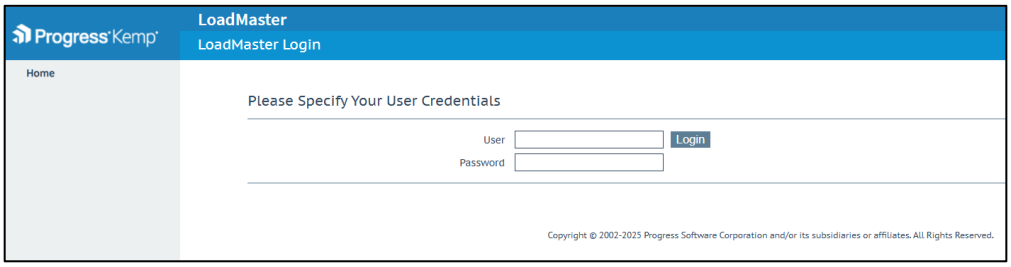
The evaluator shall confirm that the guidance documentation contains instructions for establishing the remote administrative sessions for each supported method.

Evaluator Findings:
The evaluator confirmed that the guidance documentation contains instructions for establishing the remote administrative sessions for each supported method. The relevant information is found in the following section(s): **Disable SSH Access** Upon investigation, the evaluator found that the AGD states that: **In the left frame menu, click Certificates & Security > Remote Access, and disable the Allow Remote SSH Access check box.** The relevant information is found in the following section(s): **Login Via WUI** Upon investigation, the evaluator found that the AGD states that: **Prior to authentication, the TSF only permits users to access a warning banner (if configured) and to generate system keys automatically**

during first-time setup. The TOE does not expose any other interface or administrative access prior to successful login. All authenticated access requires a valid username and password.

The following steps describe the login process via the WUI:

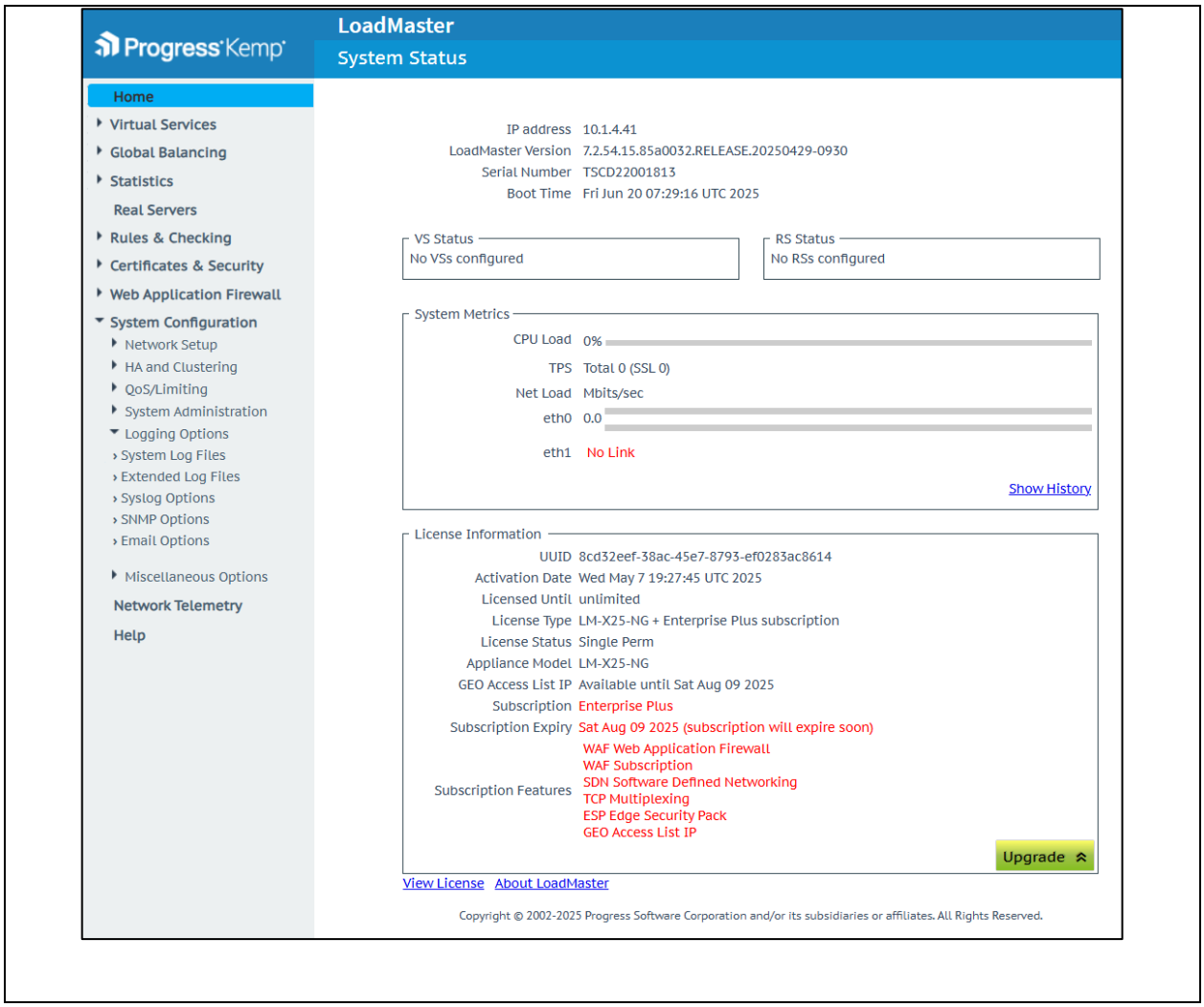
1. **Launch a web browser from a laptop that is connected to the device.**
2. **Log in to the WUI via HTTPS using the IP address assigned during installation, the 'bal' administrative login, and the password you specified during installation.**



   a. **Download the LoadMaster issuing CA ECDSA certificate and install it in the management workstation certificate store and/or the browser certificate store.**

3. **After initially logging in to the LoadMaster, if Session Management is enabled - some login information is displayed:**
   a. **The last login time and IP address of the current user**
   b. **The number of successful logins by the current user in the last 30 days**
   c. **The total number of failed logins attempts by any user (including unknown usernames) since the last successful login**
   d. **Audit Logs are generated for each action which is performed by a user; either using the API or the WUI.**
4. **The primary user (bal) always has full permissions. Secondary users may be restricted to certain functions.**
5. **Only the bal user is permitted to set the Basic Authentication password.**
6. **Users must be authenticated before logging on to the LoadMaster.**

After successful login, the Web User Interface displays the Home page with system information, as shown below:



Verdict:

PASS.

5.2 OPTIONAL REQUIREMENTS

5.2.1 SECURITY AUDIT (FAU)

5.2.1.1 FAU_STG_EXT.3 ACTION IN CASE OF POSSIBLE AUDIT DATA LOSS

5.2.1.1.1 FAU_STG_EXT.3 TSS

The evaluator shall examine the TSS to ensure that it details how the Security Administrator is warned before the local storage for audit data is full.

Evaluator Findings:
The evaluator examined the TSS and ensured that it details how the Security Administrator is warned before the local storage for audit data is full. The relevant information is found in the following section(s): TOE Summary Specification **FAU_STG_EXT.3 ACTION IN CASE OF POSSIBLE AUDIT DATA LOSS (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF generates a log when audit storage reaches 85% of capacity to inform the administrator that audit storage is nearing capacity.** **The Disk Usage under System Configuration > System Administration > System Log Files provides a visual indication of the percentage used/free of the log partition. Color coding is also used to highlight different usage levels:** • **0% to 50%: green** • **50% to 90%: orange** • **90% to 100%: red**

For distributed TOEs the evaluator shall examine the TSS to ensure it describes to which TOE components this SFR applies and how each TOE component realises this SFR. Since this SFR is optional, it might only apply to some TOE components but not all. This might lead to the situation where all TOE components store their audit information themselves but FAU_STG_EXT.3/LocSpace is supported only by one of the components. In particular, the evaluator has to verify, that the TSS describes for every component supporting this functionality, whether the warning is generated by the component itself or through another component and name the corresponding component in the latter case. The evaluator has to verify that the TSS makes clear any situations in which audit records might be 'invisibly lost'.

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

Verdict:

PASS.

5.2.1.1.2 FAU_STG_EXT.3 AGD

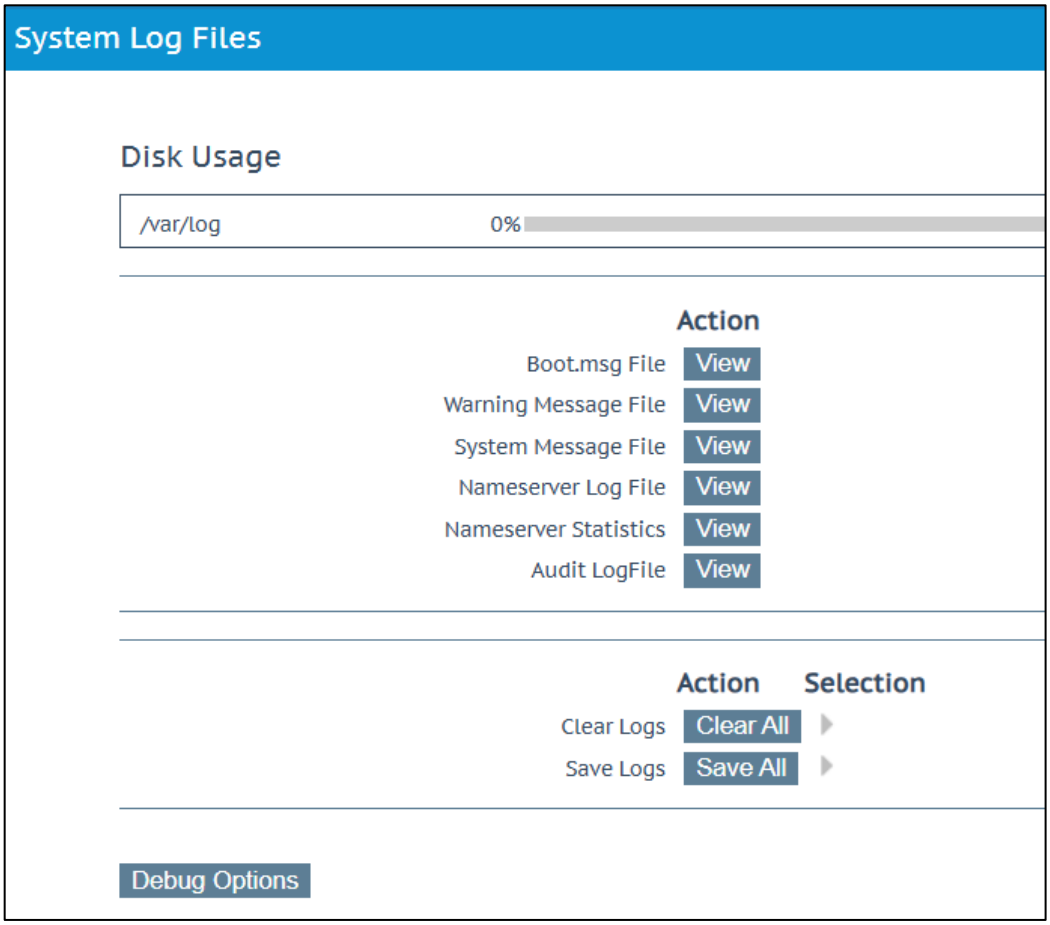
The evaluator shall also ensure that the guidance documentation describes how the Security Administrator is warned before the local storage for audit data is full and how this warning is displayed or stored (since there is no guarantee that an administrator session is running at the time the warning is issued, it is probably stored in the log files). The description in the guidance documentation shall correspond to the description in the TSS.

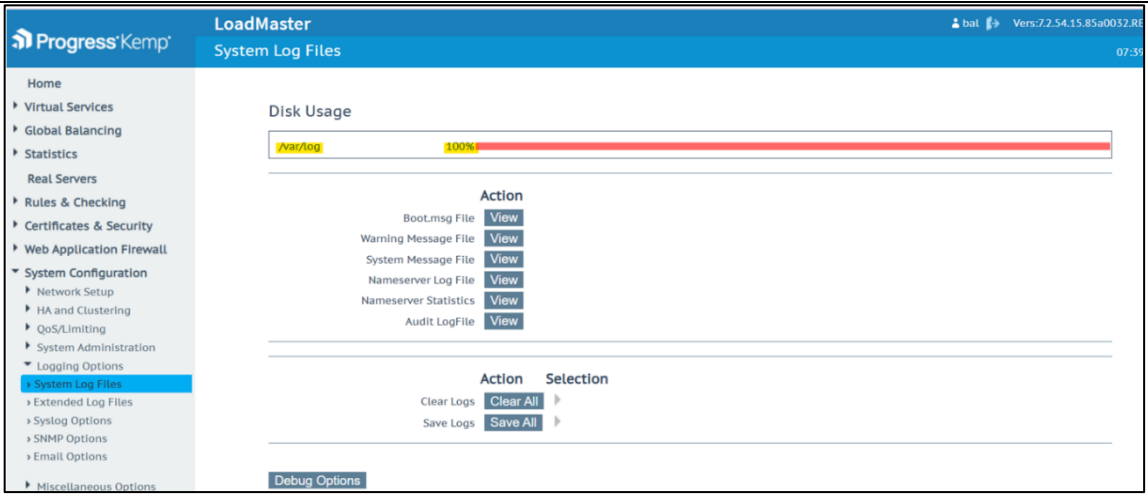
Evaluator Findings:
The evaluator examined the AGD and ensured that it describes how the Security Administrator is warned before the local storage for audit data is full and how this warning is displayed or stored. The description in the AGD corresponds to the description in the TSS. The relevant information is found in the following section(s): **AUDITING** Upon investigation, the evaluator found that the AGD states that: **The Progress LoadMaster generates a warning log when audit storage reaches 85% of capacity to inform the administrator that audit storage is nearing capacity.**

The Disk Usage under System Configuration > System Administration > System Log Files provides a visual indication of the percentage used/free of the log partition. Color-coding is used to highlight different usage levels:

- **0% to 50%: green**
- **50% to 90%: orange**
- **90% to 100%: red**

The administrator can view locally stored logs using the Web User Interface by navigating to: System Configuration > Logging Options > System Log Files





Verdict:

PASS.

5.3 SELECTION-BASED REQUIREMENTS

5.3.1 CRYPTOGRAPHIC SUPPORT (FCS)

5.3.1.1 FCS_HTTPS_EXT.1 HTTPS PROTOCOL

5.3.1.1.1 FCS_HTTPS_EXT.1 TSS

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

Evaluator Findings:
The evaluator examined the TSS and determine that enough detail is provided to explain how the implementation complies with RFC 2818. The relevant information is found in the following section(s): TOE Summary Specification **FCS_HTTPS_EXT.1 HTTPS PROTOCOL (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF acts as an HTTPS server to secure administrative connections to the WUI. The TSF implements HTTPS as specified in RFC 2818 using TLS as specified in FCS_TLSS_EXT.1. The TOE's HTTPS protocol complies with RFC 2818. The TOE implements all "MUST", "REQUIRED", and "SHOULD" statements from the RFC 2818 that are applicable to a HTTP server. The TOE web GUI operates on an explicit TCP port designed to natively implement TLS. The web server attempts to send closure Alerts prior to closing a connection in accordance with section 2.2.2 of RFC 2818.**

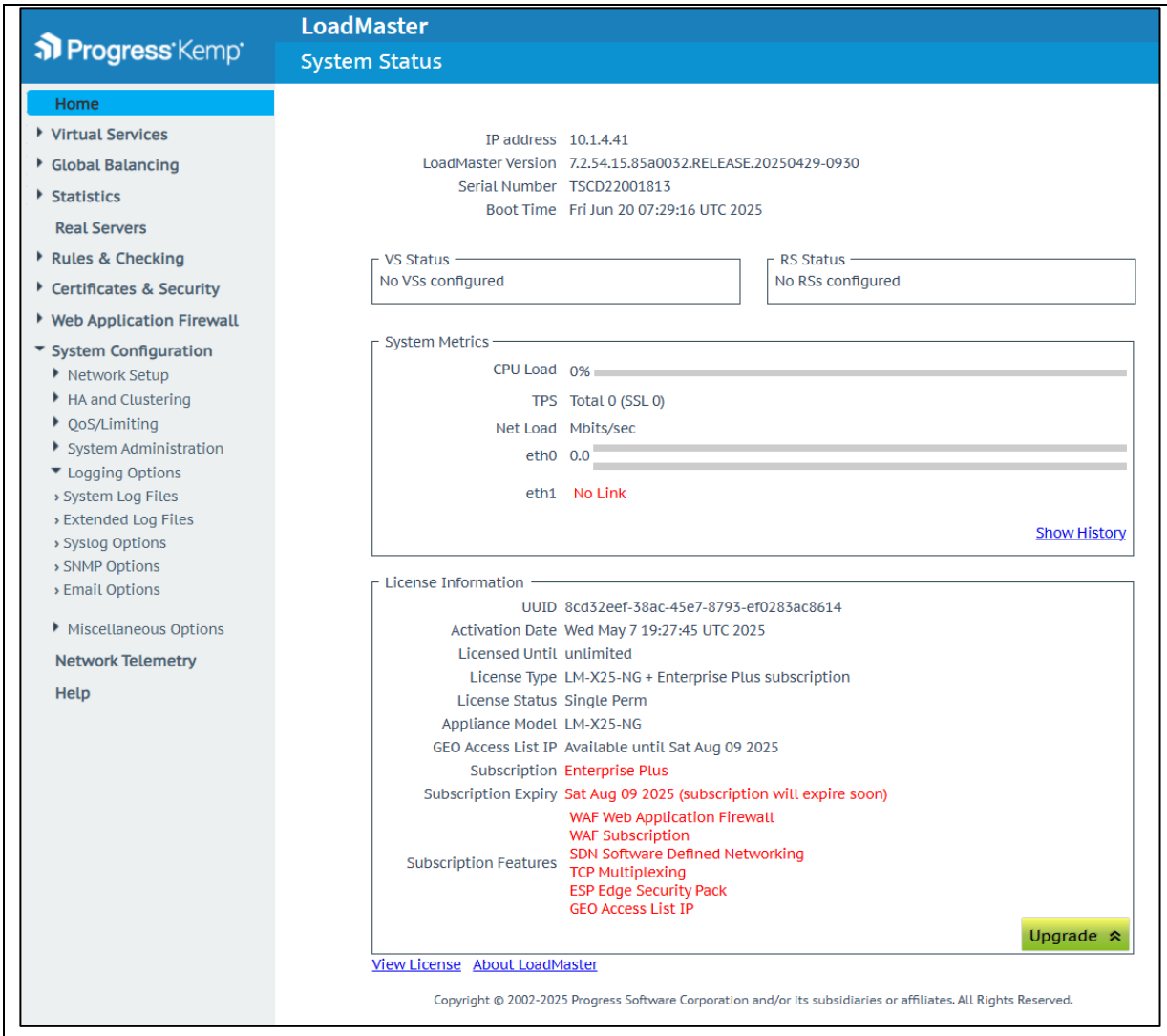
Verdict:

PASS.

5.3.1.1.2 FCS_HTTPS_EXT.1 AGD

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

Evaluator Findings:
The evaluator examined the guidance documentation to verify it instructs the Administrator how to configure TOE for use as an HTTPS client or HTTPS server. The relevant information is found in the following section(s): **Login via WUI** Upon investigation, the evaluator found that the AGD states that: **Prior to authentication, the TSF only permits users to access a warning banner (if configured) and to generate system keys automatically during first-time setup. The TOE does not expose any other interface or administrative access prior to successful login. All authenticated access requires a valid username and password.** **The following steps describe the login process via the WUI:** **1. Launch a web browser from a laptop that is connected to the device.** **2. Log in to the WUI via HTTPS using the IP address assigned during installation, the 'bal' administrative login, and the password you specified during installation.** **a. Download the LoadMaster issuing CA ECDSA certificate and install it in the management workstation certificate store and/or the browser certificate store.** **3. After initially logging in to the LoadMaster, if Session Management is enabled - some login information is displayed:** **a. The last login time and IP address of the current user** **b. The number of successful logins by the current user in the last 30 days** **c. The total number of failed logins attempts by any user (including unknown usernames) since the last successful login** **d. Audit Logs are generated for each action which is performed by a user; either using the API or the WUI.** **4. The primary user (bal) always has full permissions. Secondary users may be restricted to certain functions.** **5. Only the bal user is permitted to set the Basic Authentication password.** **6. Users must be authenticated before logging on to the LoadMaster.** **After successful login, the Web User Interface displays the Home page with system information, as shown below:**



The relevant information is found in the following section(s): **TLS Server(WUI Access)**

Upon investigation, the evaluator found that the AGD states that: **The TOE acts as a TLS server to provide secure administrator access via the Web UI. It supports TLSv1.2 only.**

The following cipher suites are supported for TLS server connections:

TLS 1.2 Cipher Suites (List 1):

- **TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA (RFC 8422)**
- **TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA (RFC 8422)**
- **TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 (RFC 5289)**
- **TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 (RFC 5289)**
- **TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 (RFC 5289)**
- **TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (RFC 5289)**

Connections using TLS versions earlier than 1.2 (e.g., TLS 1.1 or SSL) are rejected.

Note: Configuration steps for all supported algorithms (e.g., ECDSA certificate, TLS versions, and cipher suites) are detailed in Section Error! Reference source not found..

Elliptic Curve Support

- **By default, TOE supports ECDHE curves: secp256r1, secp384r1, secp521r1.**
- **Connections proposing unsupported curves will fail.**

Session Management

- **Session resumption and session tickets are not supported.**

The TOE's TLS implementation does not support or negotiate the TLS early data extension.

No configuration is required by the administrator to enable reject all renegotiation attempts behavior, as the TSF enforces the rejection of renegotiation attempts by default.

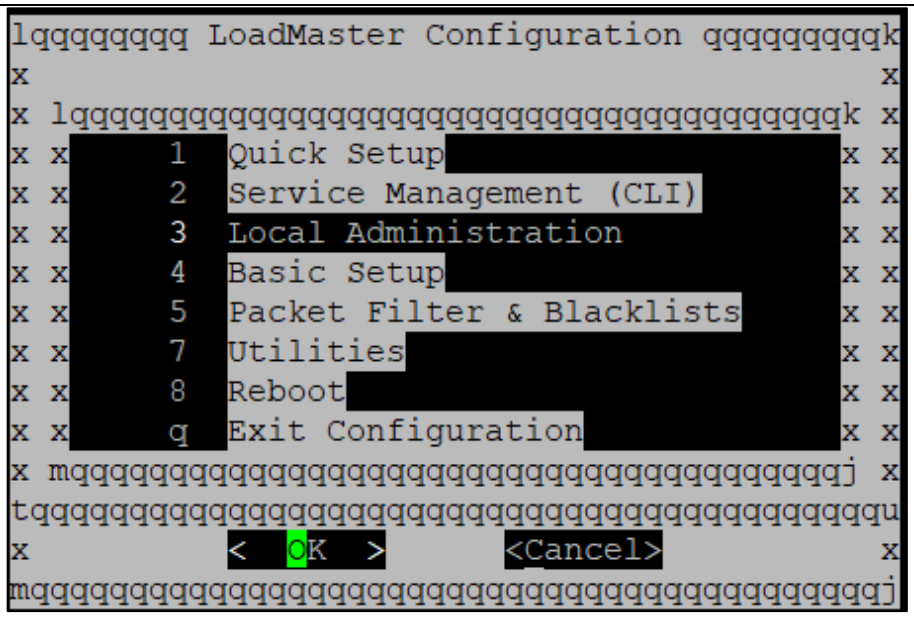
The relevant information is found in the following section(s): **START/STOP WEBUI ACCESS**

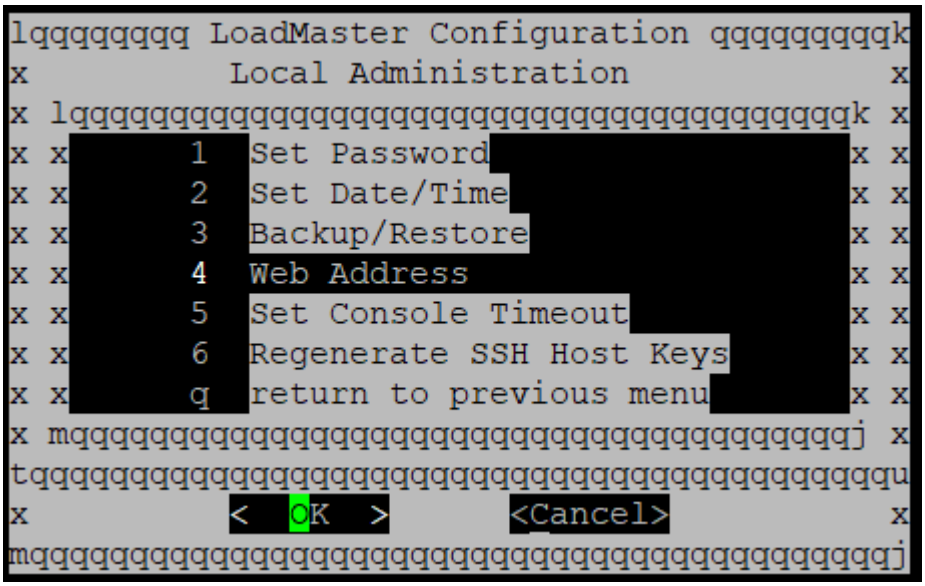
Upon investigation, the evaluator found that the AGD states that: **The TOE allows the security administrator to control access to the Web User Interface (WUI) either directly from the local console or partially through the WUI itself.**

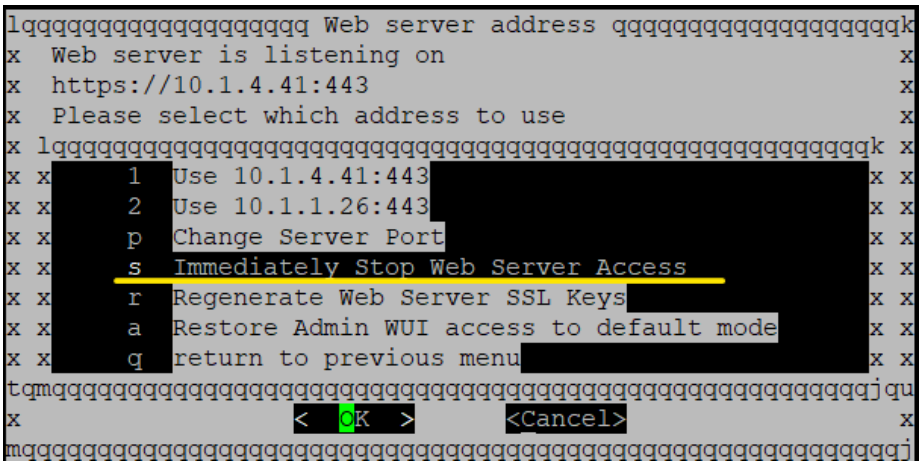
To perform this action via local console:

1. **Access the TOE via local console.**
2. **Navigate to Local Administration > Web Address > Immediately Stop/Start Web Server Access.**
3. **Use this option to:**
    - **Stop: Immediately disable access to the WUI, blocking all remote browser-based administrative access.**
    - **Start: Re-enable access to the WUI over TLS, allowing remote administration.**

The screenshot below shows the exact location of this option within the local console interface.

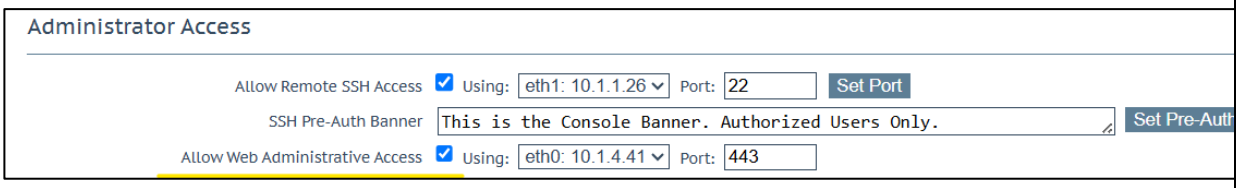

LoadMaster Configuration
1 Quick Setup
2 Service Management (CLI)
3 Local Administration
4 Basic Setup
5 Packet Filter & Blacklists
7 Utilities
8 Reboot
q Exit Configuration
< OK >
<Cancel>


LoadMaster Configuration
Local Administration
1 Set Password
2 Set Date/Time
3 Backup/Restore
4 Web Address
5 Set Console Timeout
6 Regenerate SSH Host Keys
q return to previous menu
< OK >
<Cancel>


Web server address
Web server is listening on
https://10.1.4.41:443
Please select which address to use
1 Use 10.1.4.41:443
2 Use 10.1.1.26:443
p Change Server Port
s Immediately Stop Web Server Access
r Regenerate Web Server SSL Keys
a Restore Admin WUI access to default mode
q return to previous menu
< OK >
<Cancel>

Alternatively, to disable WUI access via WUI:

1. **Navigate to Certificates & Security > Remote Access > Administrator Access.**
2. **Uncheck the Allow Web Administrative Access checkbox.**



Note: Through WUI, you can only disable WUI access. Once disabled, WUI access can only be re-enabled via the local console using the steps outlined above.

This functionality enables the administrator to manage the availability of the WUI as needed for security or maintenance purposes.

Verdict:

PASS.

5.3.1.2 FCS_NTP_EXT.1 NTP PROTOCOL

5.3.1.2.1 FCS_NTP_EXT.1.1 TSS

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

Evaluator Findings:
The evaluator examined the TSS to ensure it identifies the version of NTP supported, how it is implemented and what approach the TOE uses to ensure the timestamp it receives from an NTP timeserver (or NTP peer) is from an authenticated source and the integrity of the time has been maintained. The relevant information is found in the following section(s): TOE Summary Specification **FCS_NTP_EXT.1 NTP PROTOCOL (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF supports time updates using NTPv4. The TSF authentications updates using an administrator configured symmetric key and SHA-1.**

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

Evaluator Findings:
The TOE must support at least one of the methods or may use multiple methods, as specified in the SFR element 1.2. The evaluator shall ensure that each method selected in the ST is described in the

TSS, including the version of NTP supported in element 1.1, the message digest algorithms used to verify the authenticity of the timestamp and/or the protocols used to ensure integrity of the timestamp.

The relevant information is found in the following section(s): TOE Summary Specification **FCS_NTP_EXT.1 NTP PROTOCOL (CPP_ND_V3.0E)**

Upon investigation, the evaluator found that the TSS states that: **The TSF authentications updates using an administrator configured symmetric key and SHA-1. The TOE rejects broadcast and multicast time updates. The TOE allows up to 10 NTP time sources to be configured.**

Verdict:

PASS.

5.3.1.2.2 FCS_NTP_EXT.1.2, FCS_NTP_EXT.1.3, AND FCS_NTP_EXT.1.4 TSS

None.

5.3.1.2.3 FCS_NTP_EXT.1.1 AGD

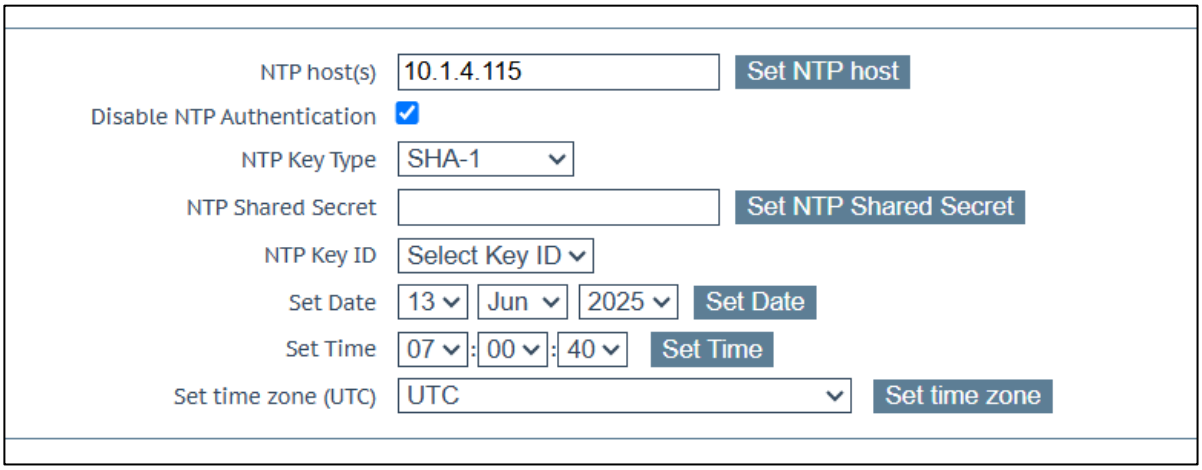
The evaluator shall examine the AGD to ensure it provides the Security Administrator instructions as how to configure the version of NTP supported, how to configure multiple NTP servers for the TOE's time source and how to configure the TOE to use the method(s) that are selected in the ST.

Evaluator Findings:

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

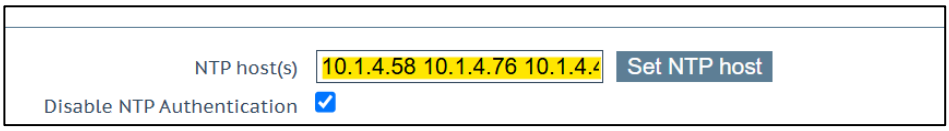
The relevant information is found in the following section(s): **CONFIGURING NTP SERVER**

Upon investigation, the evaluator found that the AGD states that: **You can manually configure the date and time of LoadMaster or leverage an NTP server. It supports time updates using NTPv4 only. The LoadMaster authentications updates using an administrator configured symmetric key and SHA-1. By default, the TOE rejects broadcast and multicast time updates. The TOE allows up to 10 NTP time sources to be configured.**



In the left frame menu, click System Configuration > System Administration > Date/Time:

a. NTP host(s): Specify the IP address of the NTP server to be used. You may configure multiple NTP servers by entering a space-separated list of hosts.



b. Show NTP Authentication Parameters: Enable the Show NTP Authentication Parameters check box to display the parameters that are needed to support NTP authenticated requests. Disable the Show NTP Authentication Parameters checkbox to hide.

c. NTP Key Type: Select SHA-1 from the drop-down menu for the NTP key type to ensure the integrity and authenticity of the timestamp data exchanged between the TOE and the NTP server.

d. NTP Shared Secret: The NTP shared secret string. The NTP secret can be a maximum of 20 ASCII characters long or 40 hexadecimal characters long.

e. NTP Key ID: Select the NTP key ID. The values range from 1 to 99. Different key IDs can be used for different servers.

Verdict:

PASS.

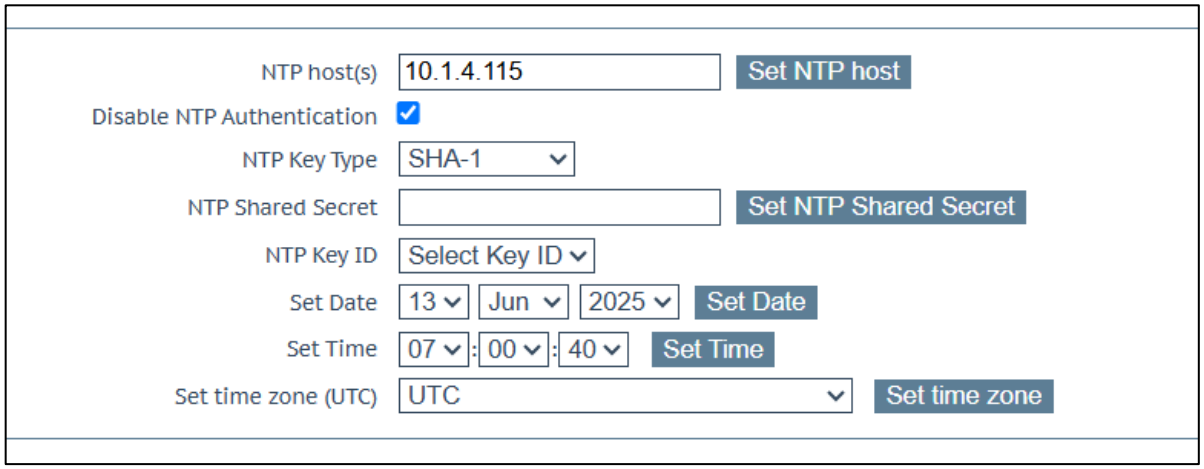
5.3.1.2.4 FCS_NTP_EXT.1.2 AGD

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

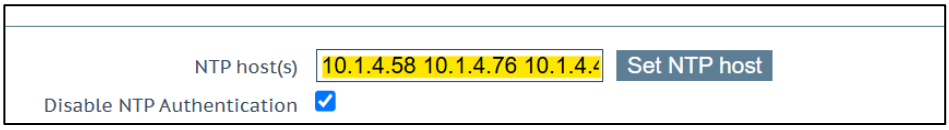
The relevant information is found in the following section(s): **CONFIGURING NTP SERVER**

Upon investigation, the evaluator found that the AGD states that:**The LoadMaster authentications updates using an administrator configured symmetric key and SHA-1.**



In the left frame menu, click System Configuration > System Administration > Date/Time:

a. NTP host(s): Specify the IP address of the NTP server to be used. You may configure multiple NTP servers by entering a space-separated list of hosts.



b. Show NTP Authentication Parameters: Enable the Show NTP Authentication Parameters check box to display the parameters that are needed to support NTP authenticated requests. Disable the Show NTP Authentication Parameters checkbox to hide.

c. NTP Key Type: Select SHA-1 from the drop-down menu for the NTP key type to ensure the integrity and authenticity of the timestamp data exchanged between the TOE and the NTP server.

d. NTP Shared Secret: The NTP shared secret string. The NTP secret can be a maximum of 20 ASCII characters long or 40 hexadecimal characters long.

e. NTP Key ID: Select the NTP key ID. The values range from 1 to 99. Different key IDs can be used for different servers.

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

Evaluator Findings:
Each primary selection in the SFR contains selections that specify a cryptographic algorithm or cryptographic protocol. For each of these secondary selections made in the ST, the evaluator examined the guidance documentation section titled **CONFIGURING NTP SERVER** and ensured that the documentation instructs the Security Administrator how to configure the TOE to use the chosen option(s).

Verdict:

PASS.

5.3.1.2.5 FCS_NTP_EXT.1.3 AGD

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

Evaluator Findings:
The evaluator examined the AGD to ensure it provides the Security Administrator instructions as how to configure the TOE to not accept broadcast and multicast NTP packets that would result in the timestamp being updated. The relevant information is found in the following section(s): **CONFIGURING NTP SERVER** Upon investigation, the evaluator found that the AGD states that: **By default, the TOE rejects broadcast and multicast time updates.**

Verdict:

PASS.

5.3.1.2.6 FCS_NTP_EXT.1.4 AGD

None.

5.3.1.3 FCS_TLSC_EXT.1 TLS CLIENT PROTOCOL

5.3.1.3.1 FCS_TLSC_EXT.1.1 TSS

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

Evaluator Findings:
The evaluator checked the description of the implementation of this protocol in the TSS and ensured that the ciphersuites supported are specified. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSC_EXT.1 TLS CLIENT PROTOCOL (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF is a TLS client for securing communications with Syslog and LDAP servers. The TSF supports TLSv1.2 (and rejects all other TLS and SSL versions) with the following ciphersuites:** • **TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 8422** • **TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 8422** • **TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289** • **TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289** • **TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289** • **TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289**

The evaluator shall check the TSS to ensure that the ciphersuites specified include those listed for this component.

Evaluator Findings:
The evaluator checked the TSS and ensured that the ciphersuites specified include those listed for this component. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSC_EXT.1 TLS CLIENT PROTOCOL (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF is a TLS client for securing communications with Syslog and LDAP servers. The TSF supports TLSv1.2 (and rejects all other TLS and SSL versions) with the following ciphersuites:** • **TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 8422** • **TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 8422** • **TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289** • **TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289** • **TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289** • **TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289** **The security administrator can configure the list of supported ciphersuites in the TOE.**

Verdict:

PASS.

5.3.1.3.2 FCS_TLSC_EXT.1.2 TSS

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

Evaluator Findings:
The evaluator ensured that the TSS describes the client's method of establishing all reference identifiers from the administrator/application- configured reference identifier, including which types of reference identifiers are supported (e.g. application-specific Subject Alternative Names) and whether IP addresses and wildcards are supported. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSC_EXT.1 TLS CLIENT PROTOCOL (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF supports the following identifier types:** • **DNS name in the SAN or CN. Wildcards are supported in the left-most position.** • **IPv4 address in the SAN or CN.** **The TSF only checks the identifier in the CN if the SAN extension is not present. The TSF does not support SRV or URI identifiers.** **The reference identifier for external IT devices are configured by the administrator using the available administrative commands in the CLI. The reference identifiers must be an IPv4 address or a hostname.** **When the reference identifier is a hostname, the TOE compares the hostname against all the DNS Name entries in the Subject Alternative Name (SAN) extension. If the hostname does not match any of the DNS Name entries, then the verification fails. If the certificate does not contain any DNS Name entries in SAN, the TOE will compare the hostname against the Common Name (CN). If the hostname does not match the CN, then the verification fails. For both DNS Name and CN matching, the hostname must be an exact match or wildcard match. In the case of a wildcard match, the wildcard must be the left-most component, wildcard matches a single component, and there are at least two non-wildcard components. When the reference identifier is an IP address, the TOE converts the IP address to a binary representation in network byte order. IPv4 addresses are converted directly from decimal to binary with period "." serving as the delineator. The TOE compares the binary IP address against all the IP Address entries in the Subject Alternative Name extension. If there is not an exact binary match, then the verification fails. If the SAN entry is missing, the TOE will compare the IPv4 address against the Common Name (CN). If the IPv4 address in CN is not an exact binary match, then the verification fails. For IPv4 address in SAN or CN matching, the IPv4 address must be an exact binary match. SAN is prioritized over CN.**

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

Evaluator Findings:
NA. The TOE is not a distributed TOE.

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

Evaluator Findings:
If IP addresses are supported in the CN as reference identifiers, the evaluator ensured that the TSS describes the TOE's conversion of the text representation of the IP address in the CN to a binary representation of the IP address in network byte order. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSC_EXT.1 TLS CLIENT PROTOCOL (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **When the reference identifier is an IP address, the TOE converts the IP address to a binary representation in network byte order. IPv4 addresses are converted directly from decimal to binary with period "." serving as the delineator. The TOE compares the binary IP address against all the IP Address entries in the Subject Alternative Name extension. If there is not an exact binary match, then the verification fails. If the SAN entry is missing, the TOE will compare the IPv4 address against the Common Name (CN). If the IPv4 address in CN is not an exact binary match, then the verification fails. For IPv4 address in SAN or CN matching, the IPv4 address must be an exact binary match. SAN is prioritized over CN.**

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

Evaluator Findings:
The evaluator also ensured that the TSS describes whether canonical format (RFC 5952 for IPv6, RFC 3986 for IPv4) is enforced. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSC_EXT.1 TLS CLIENT PROTOCOL (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TOE does not enforce canonical format.**

Verdict:

PASS.

5.3.1.3.3 FCS_TLSC_EXT.1.3 TSS

None.

5.3.1.3.4 FCS_TLSC_EXT.1.4 TSS

If "present the Supported Groups Extension" is selected, the evaluator shall verify that TSS describes the Supported Groups Extension and whether the required behaviour is performed by default or may be configured. If TLS 1.2 is claimed and DHE ciphers are claimed, then the TSS must also specify whether the TOE is capable of negotiating DHE ciphers and whether the TOE client will terminate if an

unsupported DHE parameter set is returned in the Server Key Exchange or whether all valid server-generated DHE parameters are accepted.

Evaluator Findings:
If “present the Supported Groups Extension” is selected, the evaluator verified that TSS describes the Supported Groups Extension and whether the required behaviour is performed by default or may be configured. If TLS 1.2 is claimed and DHE ciphers are claimed, then the TSS also specifies whether the TOE is capable of negotiating DHE ciphers and whether the TOE client will terminate if an unsupported DHE parameter set is returned in the Server Key Exchange or whether all valid server-generated DHE parameters are accepted The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSC_EXT.1 TLS CLIENT PROTOCOL (CPP_ND_V3.0E)** 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 TOE does not accept any Diffie-Hellman (DHE) parameters other than those explicitly supported.**

Verdict:

PASS.

5.3.1.3.5 FCS_TLSC_EXT.1.5 TSS

[Conditional]: The evaluator shall verify that TSS describes the signature_algorithms extension and whether the required behavior is performed by default or may be configured.

Evaluator Findings:
[Conditional]: The evaluator verified that TSS describes the signature_algorithms extension and whether the required behavior is performed by default or may be configured. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSC_EXT.1 TLS CLIENT PROTOCOL (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TOE presents the signature_algorithms extension in the ClientHello message without any Administrator intervention.** **The TSF supports the signature_algorithms_extension as defined in TLS 1.2, this extension is included in the ClientHello message with the following list of signature algorithms:** • **ecdsa_secp256r1 with sha256** • **ecdsa_secp384r1 with sha384** • **ecdsa_secp521r1 with sha512** **This behavior is performed by default and need not to configure.**

[Conditional]: The evaluator shall verify that TSS describes the signature_algorithms_cert extension and whether the required behavior is performed by default or may be configured.

Evaluator Findings:
NA. The ST does not select “signature_algorithm_cert extension”.

Verdict:

PASS.

5.3.1.3.6 FCS_TLSC_EXT.1.6 TSS

The evaluator shall verify that TSS describes whether the list of supported ciphersuites can be configured or not.

Evaluator Findings:
The evaluator verified that TSS describes whether the list of supported ciphersuites can be configured or not. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSC_EXT.1 TLS CLIENT PROTOCOL (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The security administrator can configure the list of supported ciphersuites in the TOE.**

Verdict:

PASS.

5.3.1.3.7 FCS_TLSC_EXT.1.7 TSS

None.

5.3.1.3.8 FCS_TLSC_EXT.1.8 TSS

The evaluator shall verify in the TSS that, for TLS 1.3, the TOE shall not permit out-of-band provisioning of pre-shared keys (PSKs) in the evaluated configuration.

Evaluator Findings:
The evaluator verified in the TSS that, for TLS 1.3, the TOE shall not permit out-of-band provisioning of pre-shared keys (PSKs) in the evaluated configuration. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSC_EXT.1 TLS CLIENT PROTOCOL (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TOE does not support the out-of-band provisioning of PSKs.**

Verdict:

PASS.

5.3.1.3.9 FCS_TLSC_EXT.1.9 TSS

None.

5.3.1.3.10 FCS_TLSC_EXT.1.1 AGD

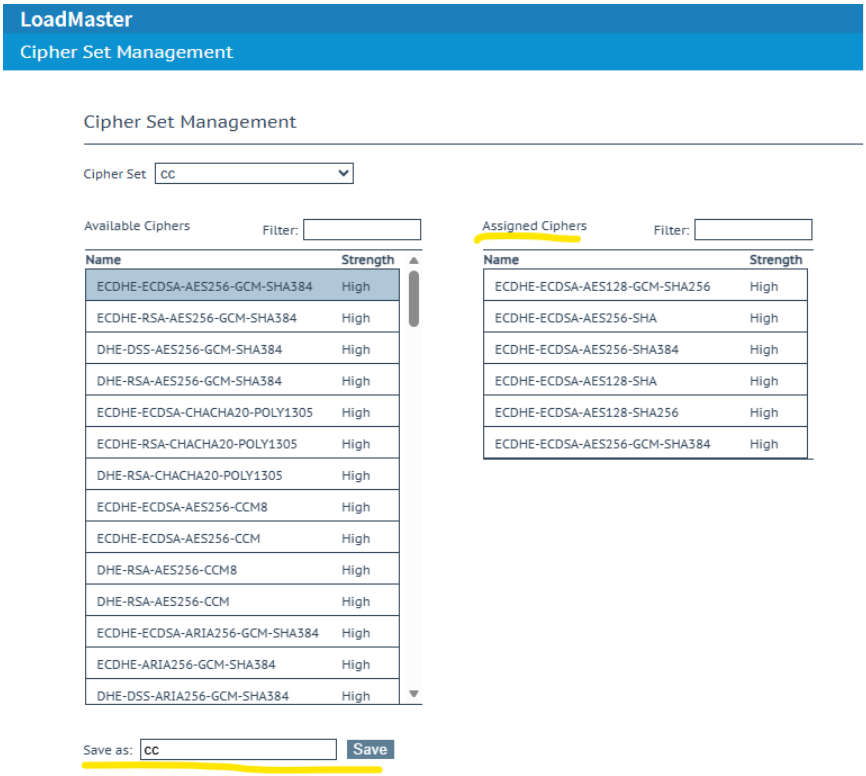
The evaluator shall check the guidance documentation to ensure that it contains instructions on configuring the TOE so that TLS conforms to the description in the TSS.

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

The relevant information is found in the following section(s): **ENABLING AND CONFIGURING TLS CIPHER SUITES FOR SECURE COMMUNICATION**

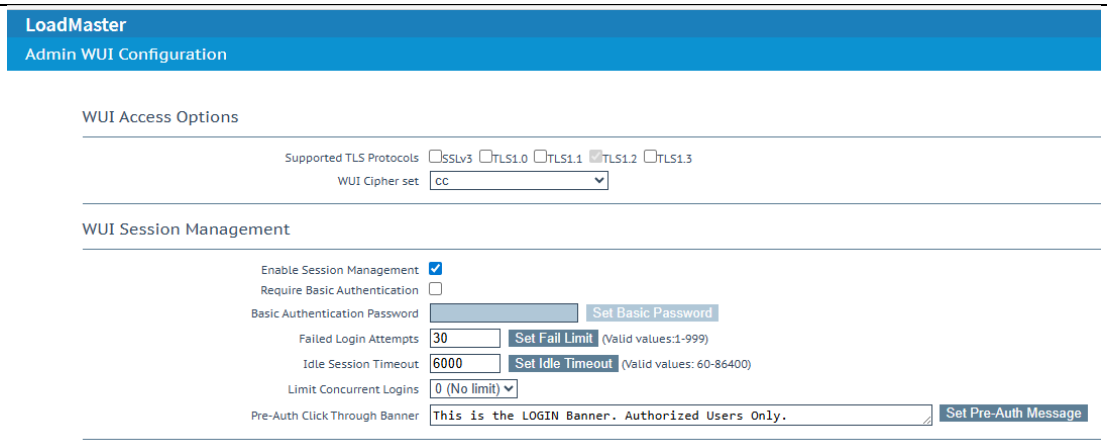
Upon investigation, the evaluator found that the AGD states that: **To configure the TLS cipher suites used by the TOE to secure WUI access, perform the following steps:**

1. **Navigate to Certificates & Security → Cipher Sets.**
2. **From the Cipher Set Management page:**
    a. **Select a Cipher Set (e.g., Default, TLS, or a custom ECC set).**
    b. **In the Available Ciphers list, select required cipher suites.**
    c. **Use the right arrow to assign selected ciphers to the Assigned Ciphers list.**
    d. **Click Save to store the changes.**

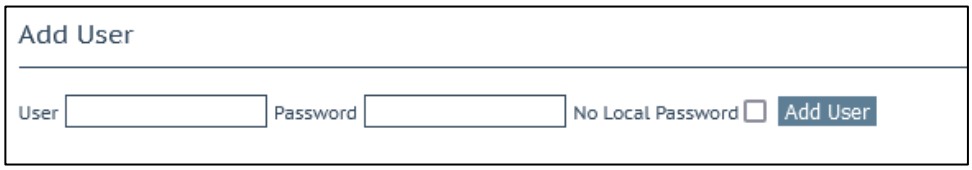


3. **Apply Cipher Set to WUI Access**
    a. **Go to Certificates & Security → Admin WUI Access:**
        - **Under WUI Access Options, select supported TLS versions (TLS 1.2).**
        - **In the WUI Cipher Set dropdown, select the configured Cipher Set.**
        - **Configure a Pre-Auth Click Through Banner, required for certificate-based authentication.**

Note: The configured cipher set is applied both for TLS server functionality (WUI access) and TLS client functionality (e.g., Syslog/LDAP).



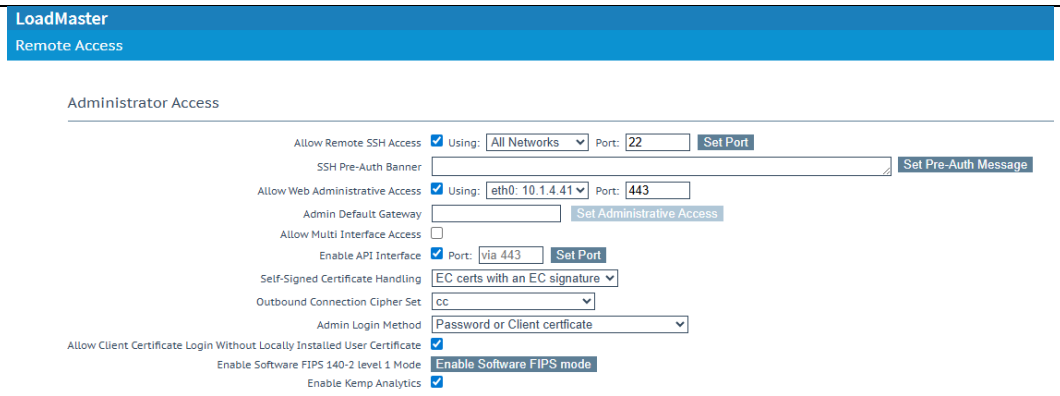
4. **Configure Client Certificate-Based Authentication**
    a. **Go to System Configuration → System Administration → User Management:**
        - **Create a user with a name that exactly matches the certificate's Principal Name.**
        - **Do not set a password; select the "no password" option.**
        - **Assign appropriate privileges (e.g., “All Rights” for the first certificate user).**



    b. **Go to Certificates & Security → Remote Access:**
        - **Set the Admin Login field to “Password or Client Certificate”.**
        - **In the Outbound Connection Cipher Set dropdown, select the configured Cipher Set.**

If login fails after configuration, try:

- **Clearing cookies and restarting the browser.**
- **Bypassing certificate request and logging in using the default bal account.**



The relevant information is found in the following section(s): TLS CLIENT (OUTBOUND CONNECTIONS TO SYSLOG/LDAP)

Upon investigation, the evaluator found that the AGD states that: TOE also acts as a TLS client for establishing outbound secure connections to trusted IT entities (e.g., Syslog or LDAP servers). These connections use the same configured cipher set as the server role.

Supported Protocols and Cipher Suites

Cipher Suites:

- **TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA (RFC 8422)**
- **TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA (RFC 8422)**
- **TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 (RFC 5289)**
- **TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 (RFC 5289)**
- **TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 (RFC 5289)**
- **TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (RFC 5289)**

Verdict:

PASS.

5.3.1.3.11 FCS_TLSC_EXT.1.2 AGD

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

Evaluator Findings:
The evaluator ensured that the AGD describes all supported identifiers, explicitly states whether the TOE supports the SAN extension or not and includes detailed instructions on how to configure the reference identifier(s) used to check the identity of peer(s). If the identifier scheme implemented by

the TOE includes support for IP addresses, the evaluator ensured that the AGD provides a set of warnings and/or CA policy recommendations that would result in secure TOE use.
The relevant information is found in the following section(s): **Reference Identifiers**
Upon investigation, the evaluator found that the AGD states that:

- **The reference identifier for external IT entities is configured by the administrator using the available administrative commands in the CLI. The reference identifiers must be an IPv4 address, or a hostname.**
- **When the reference identifier is a hostname, the TOE compares the hostname against all the DNS Name entries in the Subject Alternative Name (SAN) extension. If the hostname does not match any of the DNS Name entries, then the verification fails. If the certificate does not contain any DNS Name entries in SAN, the TOE will compare the hostname against the Common Name (CN). If the hostname does not match the CN, then the verification fails. For both DNS Name and CN matching, the hostname must be an exact match or wildcard match. In the case of a wildcard match, the wildcard must be the left-most component, wildcard matches a single component, and there are at least two non-wildcard components.**
- **When the reference identifier is an IP address, the TOE converts the IP address to a binary representation in network byte order. IPv4 addresses are converted directly from decimal to binary with period "." serving as the delineator. The TOE compares the binary IP address against all the IP Address entries in the Subject Alternative Name extension. If there is not an exact binary match, then the verification fails. If the SAN entry is missing, the TOE will compare the IPv4 address against the Common Name (CN). If the IPv4 address in CN is not an exact binary match, then the verification fails. For IPv4 address in SAN or CN matching, the IPv4 address must be an exact binary match.**

  Note: SAN is prioritized over CN.

  Warning: The above-mentioned reference identifier matching rules should be taken into consideration while connecting to peers or IT entities using certificates that have DNS names or IP Addresses.

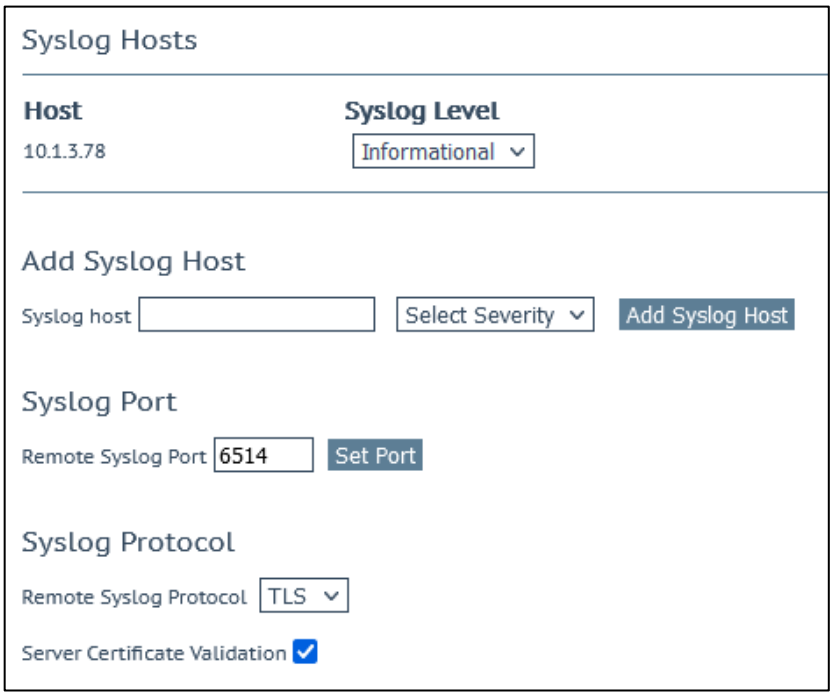
  Note: Usage of IP addresses as reference identifiers is discouraged due to their potential unreliability. As stated in RFC 6125, factors such as host mobility, NATs, and shared IPs can make IP addresses unsuitable for secure identity verification.
- **The TLS channel is terminated if verification fails.**
- **The TOE does not enforce canonical format.**

The relevant information is found in the following section(s): **Secure Remote Logging**
Upon investigation, the evaluator found that the AGD states that: **The LoadMaster can produce various warning and error messages using the syslog protocol. These messages are normally stored locally. Syslog messages are transmitted securely using TLS to remote servers. The LoadMaster uses OCSP to check the validity of the server certificates supplied by configured syslog servers. If these checks fail, connections to the server are not permitted.**

1. **In the left frame menu, click System Configuration > System Log Files > Syslog Options.**
2. **By entering the relevant IP address in the Syslog host text box and select the severity and click Add Syslog Host.**

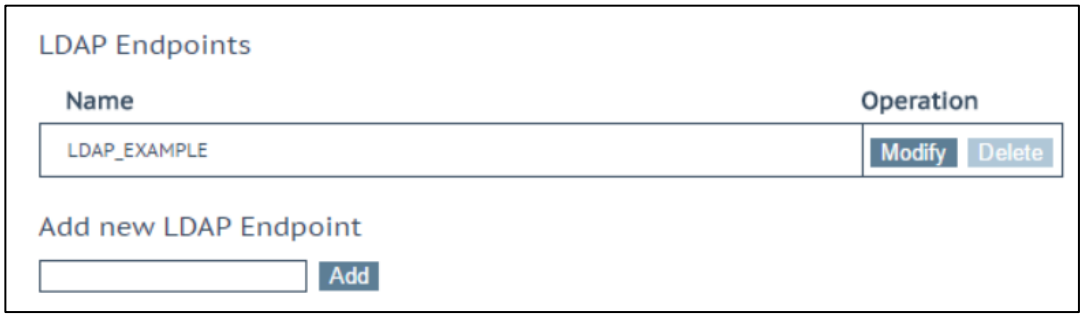
3. **Also enter the Syslog Port enter *any port other than 601* and click on Set Port.**
4. **Select the Syslog Protocol either TCP, UDP or TLS.**
5. **Enable the Server Certificate Validation.**

Note: secure syslog channel is restricted to TLSv1.2



The relevant information is found in the following section(s): **Setup Admin WUI Access Via LDAP**

Upon investigation, the evaluator found that the AGD states that: **To set up an LDAP domain, click Certificates & Security > LDAP Configuration.**

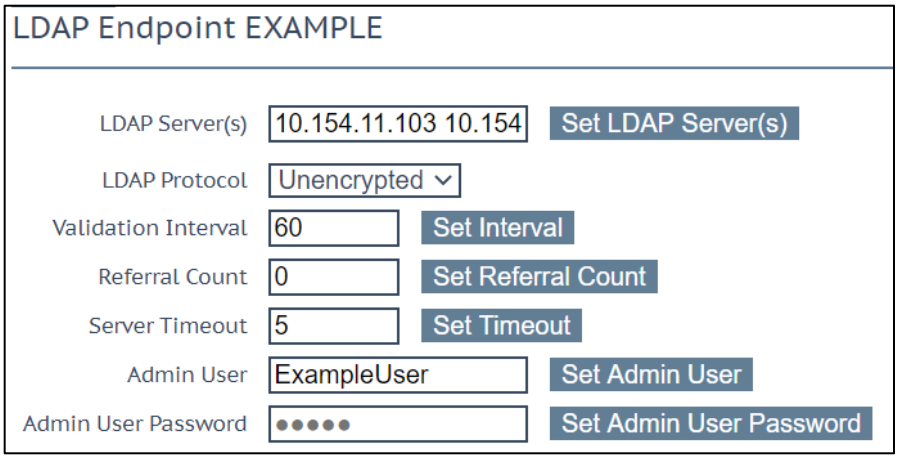


This screen provides a management interface for LDAP endpoints. These LDAP endpoints may be used in three different areas:

- **Health checks**
- **SSO domains**
- **WUI authentication**

Any existing LDAP Endpoints are listed here, with an option to Modify and Delete. If an LDAP endpoint is in use, it cannot be deleted.

There is also an option to add a new LDAP endpoint. Type a name for the endpoint and click Add. Spaces and special characters are not permitted in the LDAP endpoint name.



- **LDAP Server(s)**

  Specify a space-separated list of LDAP servers to be used. Port numbers can also be specified if required. If you have multiple domains and are using Permitted Groups, sometimes it is necessary to include the Global Catalog port number, otherwise the Permitted Groups will fail. The default port is 3628. For example, 10.110.20.23:3268.

  The LoadMaster uses OCSP to check the validity of the server certificates supplied by configured LDAPS servers. If these checks fail, connections to the server are not permitted.

- **LDAP Protocol**

  Select the transport protocol to use when communicating with the LDAP server.

  If you create an SSO domain with the Authentication Protocol set to Certificates, ensure to set the LDAP Protocol to LDAPS in the LDAP endpoint.

- **Validation Interval**

  Specify how often you should revalidate the user with the LDAP server.

- **Referral Count**

  The LoadMaster offers beta functionality to support LDAP referral replies from Active Directory Domain Controllers. If this is set to 0, referral support is not enabled. Set this field to a value between 1 and 10 to enable referral chasing. The number specified will limit the number of hops (referrals chased).

- **Server Timeout**

Specify the LDAP server timeout in seconds. The default value is 5. Valid values range from 5 to 60.

- **Admin User**

  Type the username of an administrator user.

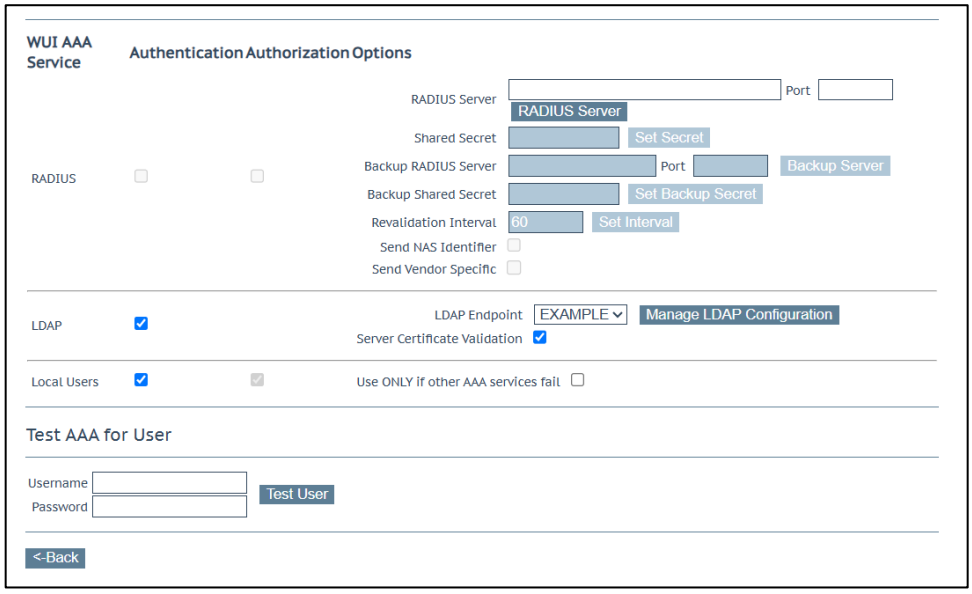
- **Admin User Password**

  Type the password for the specified administrator user.

To set up admin WUI (bal account) access via LDAP/AD:

m. In the left frame menu, click Certificates & Security > Remote Access.

n. In the page at right, click the WUI Authorization Option button.



o. Select the LDAP option.

p. Select the LDAP Endpoint that is created.

q. Add Remote User Groups if created.

r. Add the Domain and enable Server Certificate Validation option.

Note: LDAPS channel is restricted to TLSv1.2

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

Evaluator Findings:
NA. The TOE is not a distributed TOE.

Verdict:

PASS.

5.3.1.3.12 FCS_TLSC_EXT.1.3 AGD

None.

5.3.1.3.13 FCS_TLSC_EXT.1.4 AGD

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

Evaluator Findings:
If the TSS indicates that the Supported Groups Extension must be configured to meet the requirement, the evaluator verified that the AGD includes configuration of the Supported Groups Extension. The relevant information is found in the following section(s): **TLS CLIENT (OUTBOUND CONNECTIONS TO SYSLOG/LDAP)** Upon investigation, the evaluator found that the AGD states that: **By default, the TOE supports the following supported_groups extension:** • **secp256r1** • **secp384r1** • **secp521r1** **No other curves/groups are included in the extension.**

Verdict:

PASS.

5.3.1.3.14 FCS_TLSC_EXT.1.5 AGD

If the TSS indicates that the signature_algorithms extension must be configured to meet the requirement, the evaluator shall verify that AGD guidance includes configuration of the signature_algorithms extension.

Evaluator Findings:
If the TSS indicates that the signature_algorithms extension must be configured to meet the requirement, the evaluator verified that AGD guidance includes configuration of the signature_algorithms extension. The relevant information is found in the following section(s): **TLS CLIENT (OUTBOUND CONNECTIONS TO SYSLOG/LDAP)** Upon investigation, the evaluator found that the AGD states that: **By default, the TOE supports the signature_algorithms extension with:** • **ecdsa_secp256r1 with sha256**

- **ecdsa_secp384r1 with sha384**
- **ecdsa_secp521r1 with sha512**

No other signature algorithms are offered by the TOE.

Verdict:

PASS.

5.3.1.3.15 FCS_TSLC_EXT.1.6 AGD

If the TSF provides the ability of configuring the list of supported ciphersuites, the evaluator shall verify that AGD guidance includes configuration of the list of supported ciphersuites.

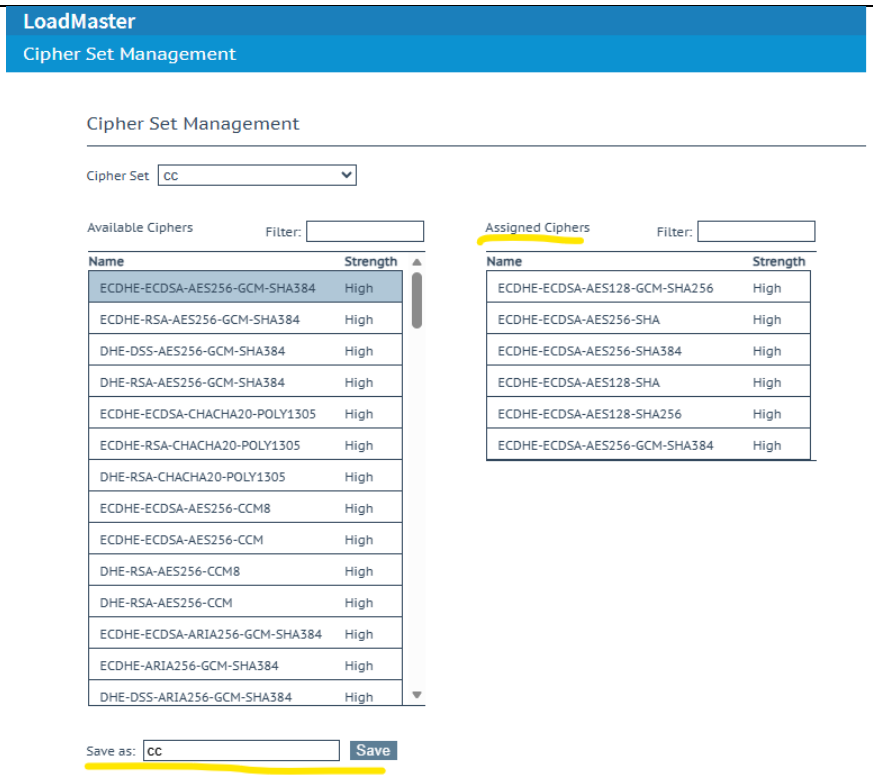
Evaluator Findings:

If the TSF provides the ability of configuring the list of supported ciphersuites, the evaluator shall verify that AGD guidance includes configuration of the list of supported ciphersuites.

The relevant information is found in the following section(s): **ENABLING AND CONFIGURING TLS CIPHER SUITES FOR SECURE COMMUNICATION**

Upon investigation, the evaluator found that the AGD states that: **To configure the TLS cipher suites used by the TOE to secure WUI access, perform the following steps:**

3. **Navigate to Certificates & Security → Cipher Sets.**
4. **From the Cipher Set Management page:**
    e. **Select a Cipher Set (e.g., Default, TLS, or a custom ECC set).**
    f. **In the Available Ciphers list, select required cipher suites.**
    g. **Use the right arrow to assign selected ciphers to the Assigned Ciphers list.**
    h. **Click Save to store the changes.**

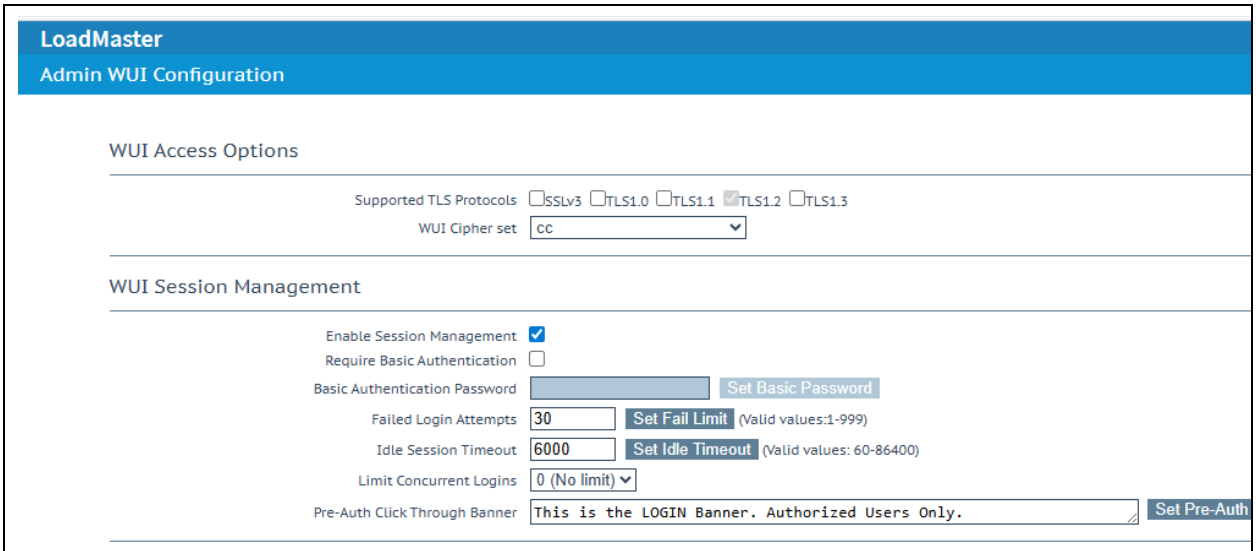


5. **Apply Cipher Set to WUI Access**

   b. **Go to Certificates & Security → Admin WUI Access:**

   - **Under WUI Access Options, select supported TLS versions (TLS 1.2).**
   - **In the WUI Cipher Set dropdown, select the configured Cipher Set.**
   - **Configure a Pre-Auth Click Through Banner, required for certificate-based authentication.**

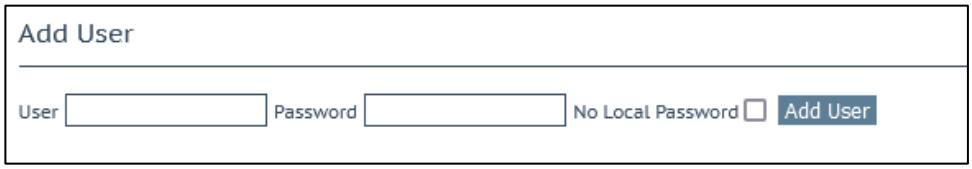
Note: The configured cipher set is applied both for TLS server functionality (WUI access) and TLS client functionality (e.g., Syslog/LDAP).



6. **Configure Client Certificate-Based Authentication**

   c. **Go to System Configuration → System Administration → User Management:**

   - **Create a user with a name that exactly matches the certificate's Principal Name.**
   - **Do not set a password; select the "no password" option.**
   - **Assign appropriate privileges (e.g., “All Rights” for the first certificate user).**

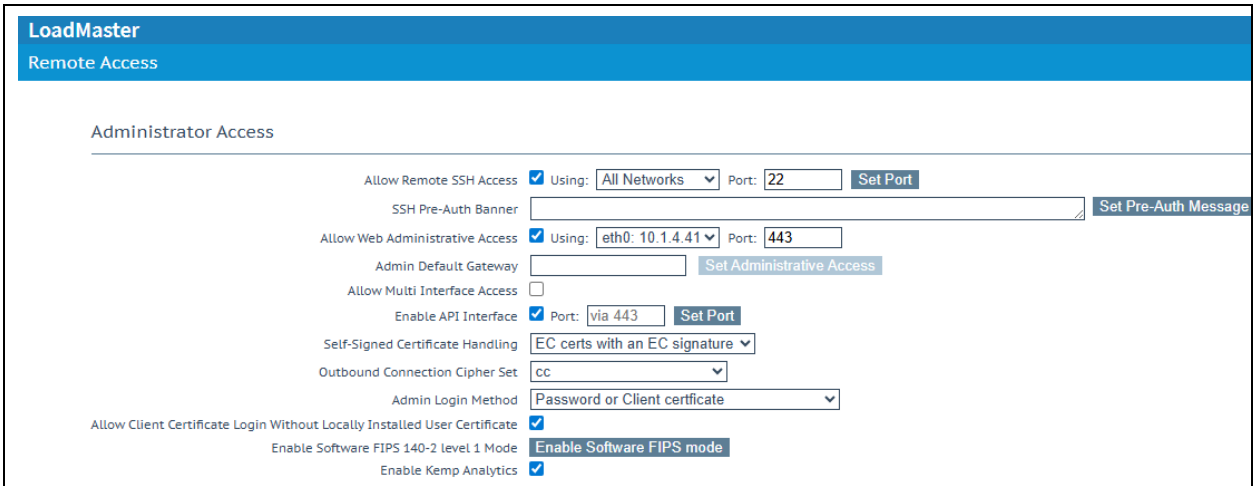


   d. **Go to Certificates & Security → Remote Access:**

   - **Set the Admin Login field to “Password or Client Certificate”.**
   - **In the Outbound Connection Cipher Set dropdown, select the configured Cipher Set.**

If login fails after configuration, try:

- **Clearing cookies and restarting the browser.**
- **Bypassing certificate request and logging in using the default bal account.**



The relevant information is found in the following section(s): TLS CLIENT (OUTBOUND CONNECTIONS TO SYSLOG/LDAP)

Upon investigation, the evaluator found that the AGD states that: TOE also acts as a TLS client for establishing outbound secure connections to trusted IT entities (e.g., Syslog or LDAP servers). These connections use the same configured cipher set as the server role.

Supported Protocols and Cipher Suites

TLSv1.2 Cipher Suites (List 1):

- **TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA (RFC 8422)**
- **TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA (RFC 8422)**
- **TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 (RFC 5289)**
- **TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 (RFC 5289)**
- **TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 (RFC 5289)**
- **TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (RFC 5289)**

Verdict:

PASS.

5.3.1.3.16 FCS_TLSC_EXT.1.7 AGD

None.

5.3.1.3.17 FCS_TLSC_EXT.1.8 AGD

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

Evaluator Findings:
The evaluator verified that any configuration necessary to meet the requirement must be contained in the AGD guidance. The relevant information is found in the following section(s): **TLS CLIENT (OUTBOUND CONNECTIONS TO SYSLOG/LDAP)** Upon investigation, the evaluator found that the AGD states that: **Pre-Shared Keys (PSKs) out-of-band provisioning is not used.**

Verdict:

PASS.

5.3.1.3.18 FCS_TLSC_EXT.1.9 AGD

None.

5.3.1.4 FCS_TLSS_EXT.1 TLS SERVER PROTOCOL

5.3.1.4.1 FCS_TLSS_EXT.1.1 TSS

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

Evaluator Findings:
The evaluator checked the description of the implementation of this protocol in the TSS and ensured that the ciphersuites supported are specified. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSS_EXT.1** Upon investigation, the evaluator found that the TSS states that: **The TSF is an HTTPS/TLS server for providing the WUI trusted channel to remote administrators. The TSF supports TLSv1.2 with the following ciphersuites:** • **TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 8422** • **TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 8422** • **TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289** • **TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289** • **TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289** • **TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289** **The security administrator can configure the list of supported cipher suites in the TOE.**

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

Evaluator Findings:
The evaluator checked the TSS and ensured that the ciphersuites specified are identical to those listed for this component. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSS_EXT.1** Upon investigation, the evaluator found that the TSS states that: **The TSF is an HTTPS/TLS server for providing the WUI trusted channel to remote administrators. The TSF supports TLSv1.2 with the following ciphersuites:** • **TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 8422** • **TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 8422** • **TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289** • **TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289** • **TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289** • **TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289** **The security administrator can configure the list of supported cipher suites in the TOE.**

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

Evaluator Findings:
The evaluator verified that the TSS contains a description of how the TOE technically prevents the use of unsupported and undefined SSL and TLS versions. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSS_EXT.1** Upon investigation, the evaluator found that the TSS states that: **The TSF rejects the connection if the client attempts to establish a connection using an older version of TLS(tls1.0, tls1.1) or SSL(ssl2.0, ssl3.0).**

Verdict:

PASS.

5.3.1.4.2 FCS_TLSS_EXT.1.2 TSS

The evaluator shall verify that the TSS describes the algorithms and key sizes the TSF supports for authenticating itself to TLS clients.

Evaluator Findings:
The evaluator verified that the TSS describes the algorithms and key sizes the TSF supports for authenticating itself to TLS clients. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSS_EXT.1** Upon investigation, the evaluator found that the TSS states that: **The TSF supports the following algorithms and key sizes for authenticating itself to TLS clients and establishing keys:** • **ECDSA over a NIST secp256r1 curve.**

The evaluator shall ensure these algorithms are consistent with the selected ciphersuites.

Evaluator Findings:
The evaluator ensured these algorithms are consistent with the selected ciphersuites. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSS_EXT.1** Upon investigation, the evaluator found that the TSS states that: **The TSF supports the following algorithms and key sizes for authenticating itself to TLS clients and establishing keys:** • **ECDSA over a NIST secp256r1 curve.** **All these algorithms are consistent with the selected ciphersuite.**

Verdict:

PASS.

5.3.1.4.3 FCS_TLSS_EXT.1.3 TSS

The evaluator shall verify that the TSS lists all EC Diffie-Hellman curves and/or Diffie-Hellman groups used in the key establishment by the TOE when acting as a TLS Server. The evaluator shall ensure these algorithms are consistent with the selected ciphersuites.

Evaluator Findings:
The evaluator verified that the TSS lists all EC Diffie-Hellman curves and/or Diffie-Hellman groups used in the key establishment by the TOE when acting as a TLS Server. The evaluator shall ensure these algorithms are consistent with the selected ciphersuites. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSS_EXT.1** Upon investigation, the evaluator found that the TSS states that: **The TSF will perform ECDHE key establishment using secp256r1, secp384r1, and secp521r1. If the client did not propose one of these curves, the connection fails.**

Verdict:

PASS.

5.3.1.4.4 FCS_TLSS_EXT.1.4 TSS

The evaluator shall verify that the TSS describes if session resumption based on session IDs is supported (RFC 4346 and/or RFC 5246), if session resumption based on session tickets is supported (RFC 5077) and/or if session resumption according to RFC 8446 is supported.

Evaluator Findings:
The evaluator shall verified that the TSS describes if session resumption based on session IDs is supported (RFC 4346 and/or RFC 5246), if session resumption based on session tickets is supported (RFC 5077) and/or if session resumption according to RFC 8446 is supported. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSS_EXT.1 TLS SERVER PROTOCOL (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF does not support session resumption or session tickets.**

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

Evaluator Findings:
NA, The TOE does not support session resumption or session tickets.

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

Evaluator Findings:
NA, The TOE does not support session resumption or session tickets.

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

Evaluator Findings:
NA, The TOE does not support session resumption or session tickets.

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

Evaluator Findings:
NA, The TOE does not support session resumption or session tickets.

Verdict:

PASS.

5.3.1.4.5 FCS_TLSS_EXT.1.5 TSS

The evaluator shall verify that TSS describes whether the list of supported ciphersuites can be configured or not.

Evaluator Findings:
The evaluator shall verify that TSS describes whether the list of supported ciphersuites can be configured or not. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSS_EXT.1 TLS SERVER PROTOCOL (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The security administrator can configure the list of supported ciphersuites in the TOE.**

Verdict:

PASS.

5.3.1.4.6 FCS_TLSS_EXT.1.6 TSS

None.

5.3.1.4.7 FCS_TLSS_EXT.1.7 TSS

The evaluator shall verify in the TSS that, for TLS 1.3, the TOE shall not permit out-of-band provisioning of pre-shared keys (PSKs) in the evaluated configuration.

Evaluator Findings:
The evaluator verified in the TSS that, for TLS 1.3, the TOE shall not permit out-of-band provisioning of pre-shared keys (PSKs) in the evaluated configuration. The relevant information is found in the following section(s): TOE Summary Specification **FCS_TLSS_EXT.1 TLS SERVER PROTOCOL (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TOE does not support the out-of-band provisioning of PSKs.**

Verdict:

PASS.

5.3.1.4.8 FCS_TLSS_EXT.1.8 TSS

None.

5.3.1.4.9 FCS_TLSS_EXT.1.1 AGD

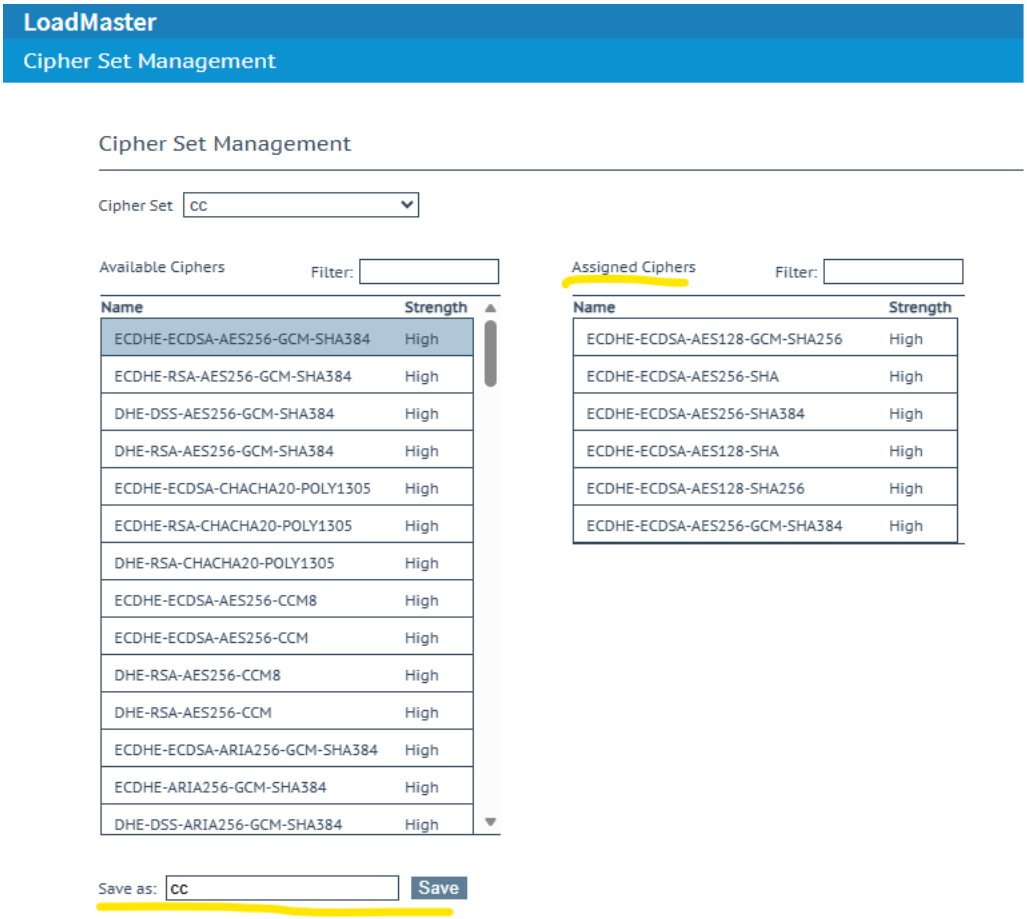
The evaluator shall check the guidance documentation to ensure that it contains instructions on configuring the TOE so that TLS conforms to the description in the TSS (for instance, the set of ciphersuites advertised by the TOE may have to be restricted to meet the requirements).

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

The relevant information is found in the following section(s): **ENABLING AND CONFIGURING TLS CIPHER SUITES FOR SECURE COMMUNICATION**

Upon investigation, the evaluator found that the AGD states that: To configure the TLS cipher suites used by the TOE to secure WUI access, perform the following steps:

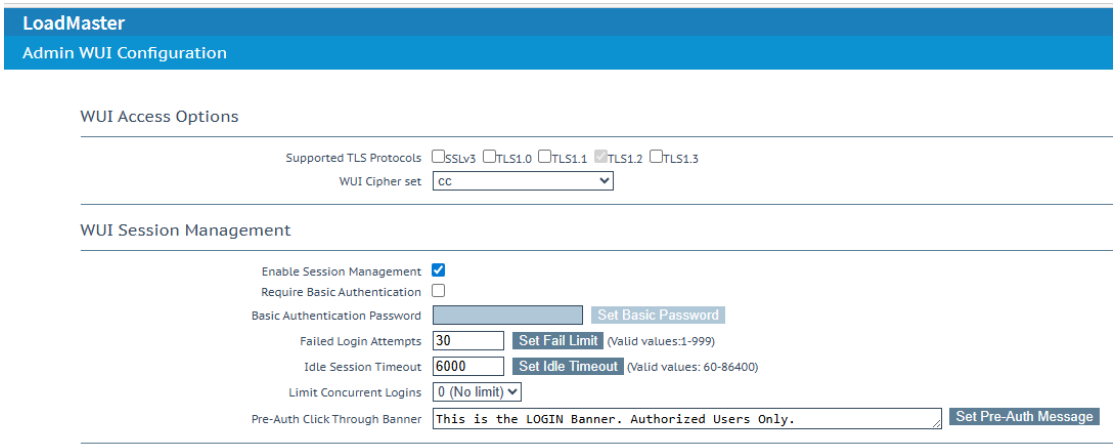
1. Navigate to **Certificates & Security → Cipher Sets**.
2. From the Cipher Set Management page:
    a. Select a Cipher Set (e.g., Default, TLS, or a custom ECC set).
    b. In the Available Ciphers list, select required cipher suites.
    c. Use the right arrow to assign selected ciphers to the Assigned Ciphers list.
    d. Click Save to store the changes.



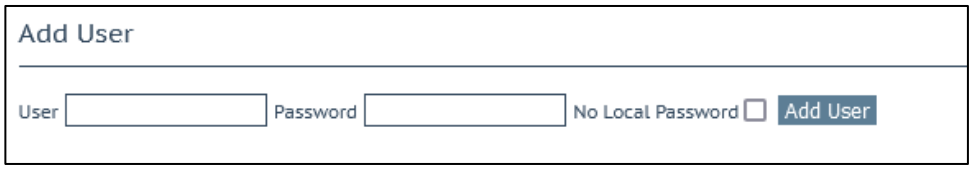
3. Apply Cipher Set to WUI Access
    a. Go to **Certificates & Security → Admin WUI Access**:
        - Under **WUI Access Options**, select supported TLS versions (TLS 1.2).
        - In the **WUI Cipher Set** dropdown, select the configured Cipher Set.

- Configure a **Pre-Auth Click Through Banner**, required for certificate-based authentication.

Note: The configured cipher set is applied both for TLS server functionality (WUI access) and TLS client functionality (e.g., Syslog/LDAP).



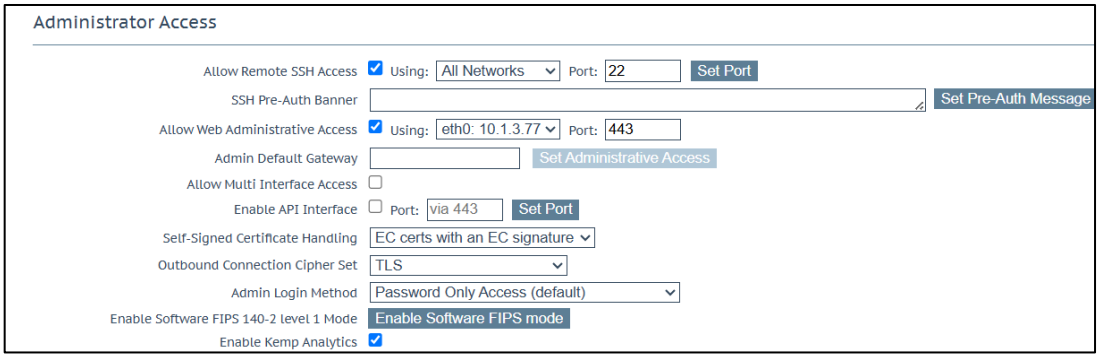
4. Configure Client Certificate-Based Authentication
    a. Go to **System Configuration → System Administration → User Management**:
        - Create a user with a name that exactly matches the certificate's Principal Name.
        - Do not set a password; select the "no password" option.
        - Assign appropriate privileges (e.g., “All Rights” for the first certificate user).



    b. Go to **Certificates & Security → Remote Access**:
        - Set the **Admin Login** field to “Password or Client Certificate”.

If login fails after configuration, try:

- Clearing cookies and restarting the browser.
- Bypassing certificate request and logging in using the default bal account.



The relevant information is found in the following section(s): **TLS SERVER (WUI ACCESS)**

Upon investigation, the evaluator found that the AGD states that: The TOE acts as a TLS server to provide secure administrator access via the Web UI. It supports TLSv1.2 only.

The following cipher suites are supported for TLS server connections:

Cipher Suites:

- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA (RFC 8422)
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA (RFC 8422)
- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 (RFC 5289)
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 (RFC 5289)
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 (RFC 5289)
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (RFC 5289)

Verdict:

PASS.

5.3.1.4.10 FCS_TLSS_EXT.1.2 AGD

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

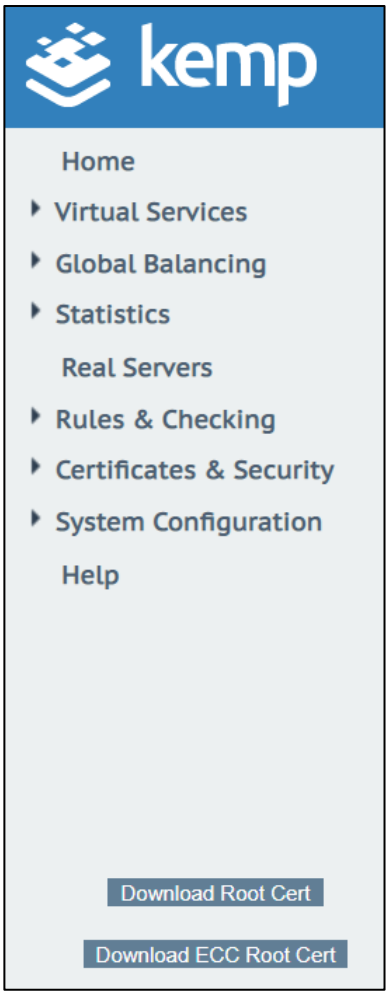
Evaluator Findings:
The evaluator verified that any configuration necessary to meet the requirement must be contained in the AGD guidance.

The relevant information is found in the following section(s): **SELF-SIGNED CERTIFICATE HANDLING (WUI)**

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

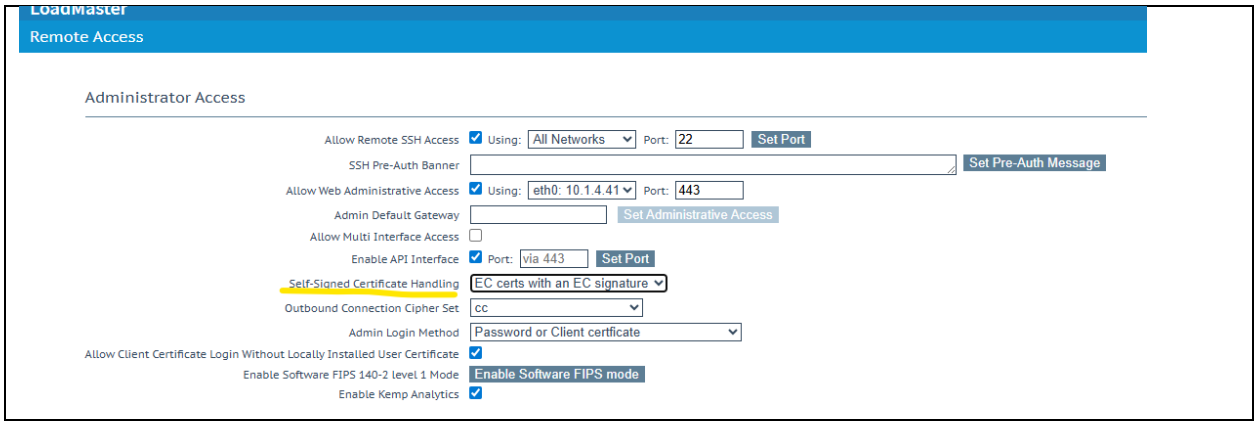
To use ECC self-signed certificates for the WUI:

- **Navigate to Certificates & Security → Remote Access.**
- **In Self-Signed Certificate Handling, choose:**
  - **EC Certs with RSA Signature (generates a new ECC certificate with RSA sig).**
    - **Download and install the ECC Issuing CA certificate on the admin workstation.**



  - **EC Certs with EC Signature (fully ECC certificate).**
    - **Important: Before selecting this, ensure the ECC Issuing CA certificate is already trusted in the admin workstation/browser.**
    - **Failing to do this will lock out WUI access, requiring console access and a factory reset (note: bal account remains unchanged post-reset).**

Note: Ensure that the server certificate is an X.509 certificate with an ECDSA public key using the secp256r1 (P-256) curve, as required by the evaluated configuration.



Verdict:

PASS.

5.3.1.4.11 FCS_TLSS_EXT.1.3 AGD

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

Evaluator Findings:
The evaluator verified that any configuration necessary to meet the requirement must be contained in the AGD guidance. The relevant information is found in the following section(s): **TLS SERVER (WUI ACCESS)** Upon investigation, the evaluator found that the AGD states that: **Elliptic Curve Support** • **By default, TOE supports ECDHE curves: secp256r1, secp384r1, secp521r1.** • **Connections proposing unsupported curves will fail.**

Verdict:

PASS.

5.3.1.4.12 FCS_TLSS_EXT.1.4 AGD

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

Evaluator Findings:
The evaluator verified that any configuration necessary to meet the requirement must be contained in the AGD guidance. The relevant information is found in the following section(s): **TLS SERVER (WUI ACCESS)** Upon investigation, the evaluator found that the AGD states that: **Session Management** • **Session resumption and session tickets are not supported.**

Verdict:

PASS.

5.3.1.4.13 FCS_TLSS_EXT.1.5 AGD

If the TSF provides the ability of configuring the list of supported ciphersuites, the evaluator shall verify that AGD guidance includes configuration of the list of supported ciphersuites.

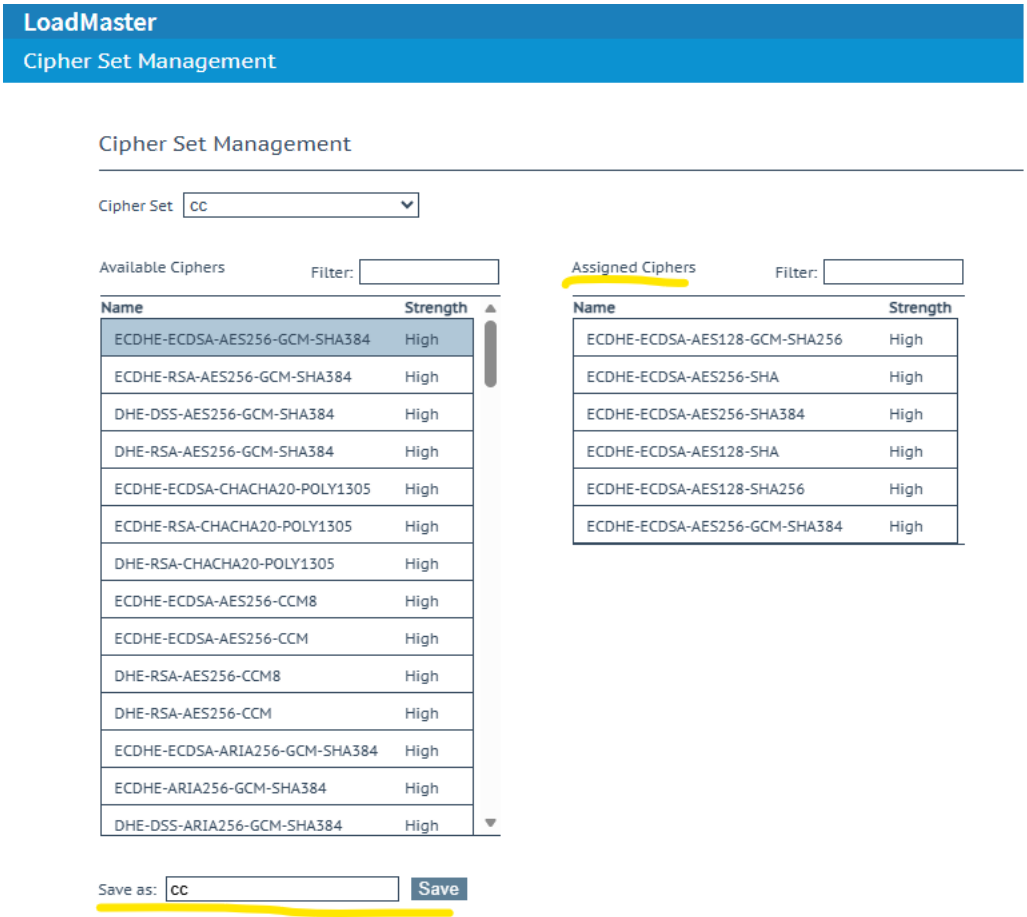
Evaluator Findings:

If the TSF provides the ability of configuring the list of supported ciphersuites, the evaluator verified that AGD guidance includes configuration of the list of supported ciphersuites.

The relevant information is found in the following section(s): **ENABLING AND CONFIGURING TLS CIPHER SUITES FOR SECURE COMMUNICATION**

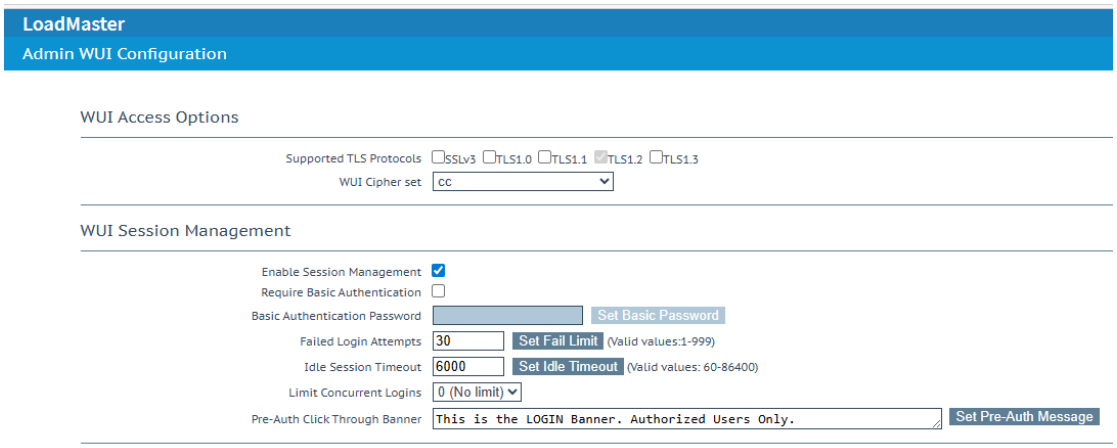
Upon investigation, the evaluator found that the AGD states that To configure the TLS cipher suites used by the TOE to secure WUI access, perform the following steps:

1. Navigate to **Certificates & Security → Cipher Sets**.
2. From the Cipher Set Management page:
    a. Select a Cipher Set (e.g., Default, TLS, or a custom ECC set).
    b. In the Available Ciphers list, select required cipher suites.
    c. Use the right arrow to assign selected ciphers to the Assigned Ciphers list.
    d. Click Save to store the changes.

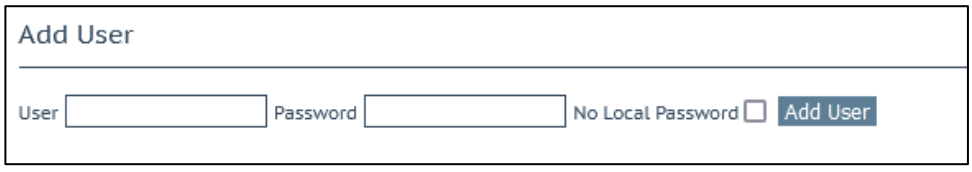


3. Apply Cipher Set to WUI Access
    a. Go to **Certificates & Security → Admin WUI Access**:
        - Under **WUI Access Options**, select supported TLS versions (TLS 1.2).
        - In the **WUI Cipher Set** dropdown, select the configured Cipher Set.
        - Configure a **Pre-Auth Click Through Banner**, required for certificate-based authentication.

Note: The configured cipher set is applied both for TLS server functionality (WUI access) and TLS client functionality (e.g., Syslog/LDAP).



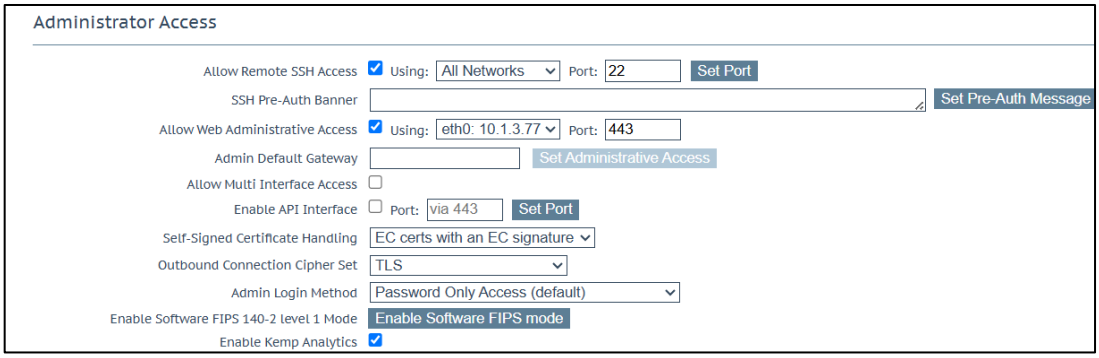
4. Configure Client Certificate-Based Authentication
    a. Go to **System Configuration → System Administration → User Management**:
        - Create a user with a name that exactly matches the certificate's Principal Name.
        - Do not set a password; select the "no password" option.
        - Assign appropriate privileges (e.g., “All Rights” for the first certificate user).



    b. Go to **Certificates & Security → Remote Access**:
        - Set the **Admin Login** field to “Password or Client Certificate”.

If login fails after configuration, try:

- Clearing cookies and restarting the browser.
- Bypassing certificate request and logging in using the default bal account.



The relevant information is found in the following section(s): **TLS SERVER (WUI ACCESS)**

Upon investigation, the evaluator found that the AGD states that: The TOE acts as a TLS server to provide secure administrator access via the Web UI. It supports TLSv1.2 only.

The following cipher suites are supported for TLS server connections:

Cipher Suites:

- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA (RFC 8422)
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA (RFC 8422)
- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 (RFC 5289)
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 (RFC 5289)
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 (RFC 5289)
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (RFC 5289)

Verdict:

PASS.

5.3.1.4.14 FCS_TLSS_EXT.1.6 AGD

None.

5.3.1.4.15 FCS_TLSS_EXT.1.7 AGD

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

Evaluator Findings:
The evaluator verified that any configuration necessary to meet the requirement must be contained in the AGD guidance. The relevant information is found in the following section(s): **TLS SERVER (WUI ACCESS)** Upon investigation, the evaluator found that the AGD states that: **The TOE does not support the out-of-band provisioning of PSKs.**

Verdict:

PASS.

5.3.1.4.16 FCS_TLSS_EXT.1.8 AGD

None.

5.3.2 IDENTIFICATION AND AUTHENTICATION (FIA)

5.3.2.1 FIA_X509_EXT.1/REV X.509 CERTIFICATE VALIDATION

5.3.2.1.1 FIA_X509_EXT.1/REV TSS

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

Evaluator Findings:
The evaluator ensured the TSS describes where the check of validity of the certificates takes place, and that the TSS identifies any of the rules for extendedKeyUsage fields (in FIA_X509_EXT.1.1) that are not supported by the TOE (i.e. where the ST is therefore claiming that they are trivially satisfied). It is expected that revocation checking is performed when a certificate is used in an authentication step and when performing trusted updates (if selected). The relevant information is found in the following section(s): TOE Summary Specification **FIA_X509_EXT.1/REV X.509 CERTIFICATE VALIDATION (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TOE performs X.509 certificate validation at the following points:** **• TOE TLS client authentication of server X.509 certificates;** **• When certificates are loaded into the TOE, such as when importing CAs and certificate responses.** **In all scenarios, certificates are checked for several validation characteristics:** **• If the certificate 'notAfter' date is in the past, then this is an expired certificate which is considered invalid;** **• The certificate chain must terminate with a certificate designated as a trust anchor on the TOE;** **• All certificates not designated as trust anchors must not be revoked, as indicated by an OCSP status check;** **• All trust anchor and intermediate certificates must contain the basicConstraints extension and have the CA flag set to TRUE.** **• Server certificates consumed by the TOE TLS client must have a 'serverAuthentication' extendedKeyUsage purpose;** **• Client certificates consumed by the TOE TLS server must have a 'clientAuthentication' extendedKeyUsage purpose;** **• Certificates used to sign OCSP responses must have the OCSP signing purpose in the extendedKeyUsage extension.** **TOE performs revocation checking when a certificate is used in an authentication step. The X.509 certificates are not used for trusted updates.** **The TOE performs revocation checking on the peer's leaf certificate and intermediate CA certificate. No different handling is applied based on the structure of the certificate chain.**

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

Evaluator Findings:
The TSS describes when revocation checking is performed and on what certificates. 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. The relevant information is found in the following section(s): TOE Summary Specification **FIA_X509_EXT.1/REV X.509 CERTIFICATE VALIDATION (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **TOE performs revocation checking when a certificate is used in an authentication step. The X.509 certificates are not used for trusted updates. The TOE performs revocation checking on the peer's leaf certificate and intermediate CA certificate. No different handling is applied based on the structure of the certificate chain.**

Verdict:

PASS.

5.3.2.1.2 FIA_X509_EXT.1/REV AGD

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

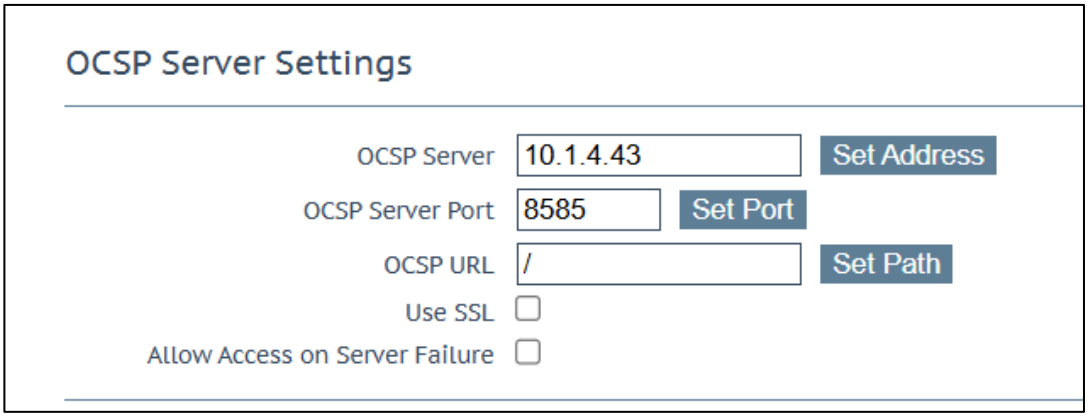
Evaluator Findings:
The evaluator also ensured that the guidance documentation describes where the check of validity of the certificates takes place, describes any of the rules for extendedKeyUsage fields (in FIA_X509_EXT.1.1) that are not supported by the TOE (i.e. where the ST is therefore claiming that they are trivially satisfied) and describes how certificate revocation checking is performed and on which certificate. The relevant information is found in the following section(s): **X509 CERTIFICATE** Upon investigation, the evaluator found that the AGD states that: **The TOE performs X.509 certificate validation at the following points:** • **TOE TLS client authentication of server X.509 certificates;** • **When certificates are loaded into the TOE, such as when importing CAs and certificate responses.** **The TOE performs certificate validation according to the following rules:** • **Certificates must not be expired. An expired certificate (where the notAfter date is in the past) is considered invalid.** • **The certificate chain must end with a trusted CA (trust anchor) that is configured on the TOE.** • **All non-trust-anchor certificates are validated for revocation status using OCSP (Online Certificate Status Protocol).** • **All CA certificates (trust anchors and intermediates) must include the basicConstraints extension with the CA flag set to TRUE.**

- **TLS server certificates must include the serverAuthentication purpose in their extendedKeyUsage field.**
- **TLS client certificates must include the clientAuthentication purpose.**
- **Certificates used to sign OCSP responses must include the OCSP Signing purpose.**

The relevant information is found in the following section(s): **ENABLE OCSP CHECKING**

In the left frame menu, click Certificates & Security > OCSP Configuration:

1. **Enter the OCSP Server IP address and click Set Address.**
2. **Enter the OCSP Server Port and click Set Port.**
3. **Enter the OCSP URL and click Set URL.**
4. **Enable the Enable OCSP Checking check box.**



5. **If the certificate cannot be validated because the server is unavailable, there is an option called Allow Access on Server Failure where you can decide if you want to pass the authentication or not. Enabling this check box treats an OCSP server connection failure or timeout as if the OCSP server has returned a valid response. That is, the client certificate is treated as valid.**

Note: The AIA is given precedence and will be used based on its presence. If the AIA is not present or appears invalid, the OCSP Server configuration details will be used.

Please note the following:

- **LDAPS: AIA information from the server certificate is honoured if Certificates & Security > OCSP Configuration > OCSP Checking is enabled.**
- **Syslog-NG: AIA information from the server certificate is honoured if Certificates & Security > OCSP Configuration > OCSP Checking is enabled.**
- **WUI Authentication: AIA information from the client certificate is honoured if the "Client certificate required (verify via OCSP)" is configured under Certificates & Security > Remote Access > Administrator Access > Admin Login Method.**

The TOE treats a certificate as a CA certificate only if the basicConstraints extension is present and the CA flag is set to TRUE. Certificates that do not contain this extension or have the CA flag set to FALSE are not treated as CA certificates and will not be accepted for certificate chaining.

As a TLS Client, the TOE performs revocation checking using the Online Certificate Status Protocol (OCSP) to determine the revocation status of peer certificates.

If the TOE does not receive an OCSP response (e.g., due to a timeout or connectivity issue), it rejects the certificate by default. However, the Security Administrator can configure the TOE to accept the certificate when no OCSP response is received, effectively allowing revocation checking to be bypassed in such cases.

Verdict:

PASS.

5.3.2.2 FIA_X509_EXT.2 X.509 CERTIFICATE AUTHENTICATION

5.3.2.2.1 FIA_X509_EXT.2 TSS

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

Evaluator Findings:
The evaluator checked the TSS and ensured that it describes how the TOE chooses which certificates to use, and any necessary instructions in the administrative guidance for configuring the operating environment so that the TOE can use the certificates. The relevant information is found in the following section(s): TOE Summary Specification **FIA_X509_EXT.2 X.509 CERTIFICATE AUTHENTICATION (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TOE provides a trust store to manage and store X.509v3 certificates used for authentication and trusted connections. Once uploaded, the TOE automatically treats the CA certificate as trust anchor no further configuration is required.**

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.

Evaluator Findings:
The evaluator examined the TSS and confirmed that it describes the behaviour of the TOE when a connection cannot be established during the validity check of a certificate used in establishing a trusted channel. The relevant information is found in the following section(s): TOE Summary Specification **FIA_X509_EXT.2 X.509 CERTIFICATE AUTHENTICATION (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **As a TLS Client, the TOE uses OCSP to determine whether the certificate is revoked or not.** **When the TSF does not receive a response from an OCSP server, by default, the TSF rejects the certificate. The administrator can configure the TSF to accept certificates when an OCSP response is not received.**

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 verified that any distinctions between trusted channels are described. If the requirement that the administrator is able to specify the default action, then the evaluator ensured that the guidance documentation contains instructions on how this configuration action is performed. The relevant information is found in the following section(s): TOE Summary Specification **FIA_X509_EXT.2 X.509 CERTIFICATE AUTHENTICATION (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The administrator can configure the TSF to accept certificates when an OCSP response is not received.**

Verdict:

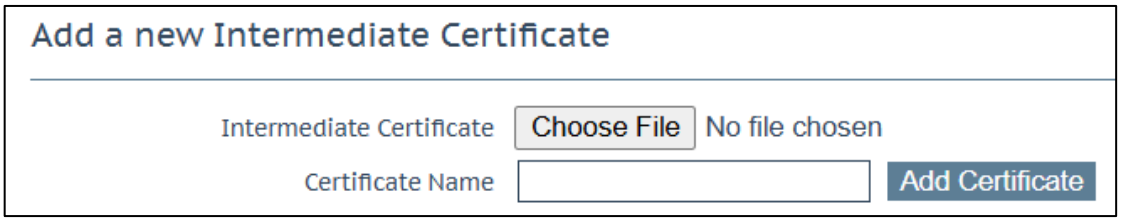
PASS.

5.3.2.2.2 FIA_X509_EXT.2 AGD

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

Evaluator Findings:
The evaluator also ensured that the guidance documentation describes the configuration required in the operating environment so the TOE can use the certificates. The guidance documentation also includes any required configuration on the TOE to use the certificates. The guidance document also describes the steps for the Security Administrator to follow if the connection cannot be established during the validity check of a certificate used in establishing a trusted channel. The relevant information is found in the following section(s): **INTERMEDIATE CERTIFICATE** Upon investigation, the evaluator found that the AGD states that: **To upload Root and Intermediate CA Certificates, before enabling certificate-based admin WUI access:** 1. **Navigate to Certificates & Security → Intermediate Certificates.**

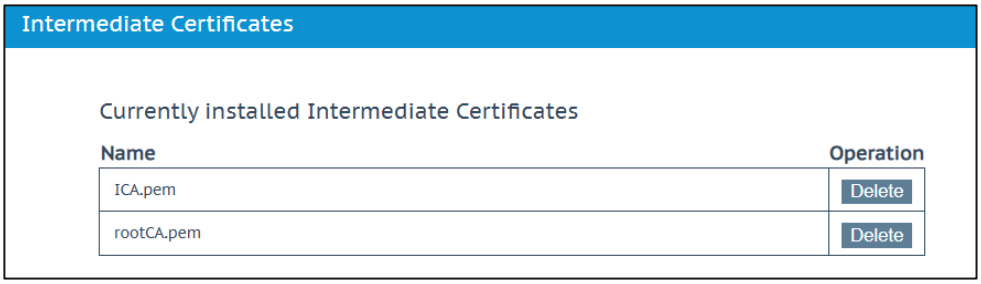
2. **Upload the issuing CA and Root CA certificates required to validate administrator certificates for client authentication.**



Note: Ensure that the server certificate is an X.509 certificate with an ECDSA public key using the secp256r1 (P-256) curve, as required by the evaluated configuration.

TSF allows the security administrator to generate cryptographic keys associated with TSF's self-signed web server certificate or Certificate Signing Requests.

The same screen as shown below displays the list of installed certificates along with their assigned names.



The relevant information is found in the following section(s): **GENERATE CSR (CERTIFICATE SIGNING REQUEST)**

Upon investigation, the evaluator found that the AGD states that: **The TOE is capable of generating a Certificate Signing Request (CSR) that includes the public key along with device-specific information such as email address, requested Subject Alternative Name (SAN), Common Name (CN), Organization (O), Organizational Unit (OU), and Country (C). The following instructions explain how to establish these fields before creating the CSR.**

- **Complete the Certificate Signing Request (CSR) form and click the Create CSR button.**
- **If Self-Signed Certificate Handling is set to EC certs with an EC signature (in Certificates & Security > Remote Access), CSR generation is restricted to the administrative (bal) user only. If Self-Signed Certificate Handling is set to a different value, all users (regardless of their permissions) can generate CSRs.**

1. **In the main menu, go to Certificates & Security > Generate CSR.**

The following are the instructions for establishing the above fields in the CSR:

- **Letter Country Code (ex. US)**

  The 2-letter country code that should be included in the certificate, for example US should be entered for the United States.

- **State/Province (Entire Name – New York, not NY)**

  The state which should be included in the certificate. Enter the full name here, for example New York, not NY.

- **City**

  The name of the city that should be included in the certificate.

- **Company (Organization)**

  The name of the organization or company which should be included in the certificate.

- **Organization Unit (e.g., Marketing, Finance, Sales)**

  The department or organizational unit that should be included in the certificate.

- **Common Name**

  The Fully Qualified Domain Name (FQDN) for your web server.

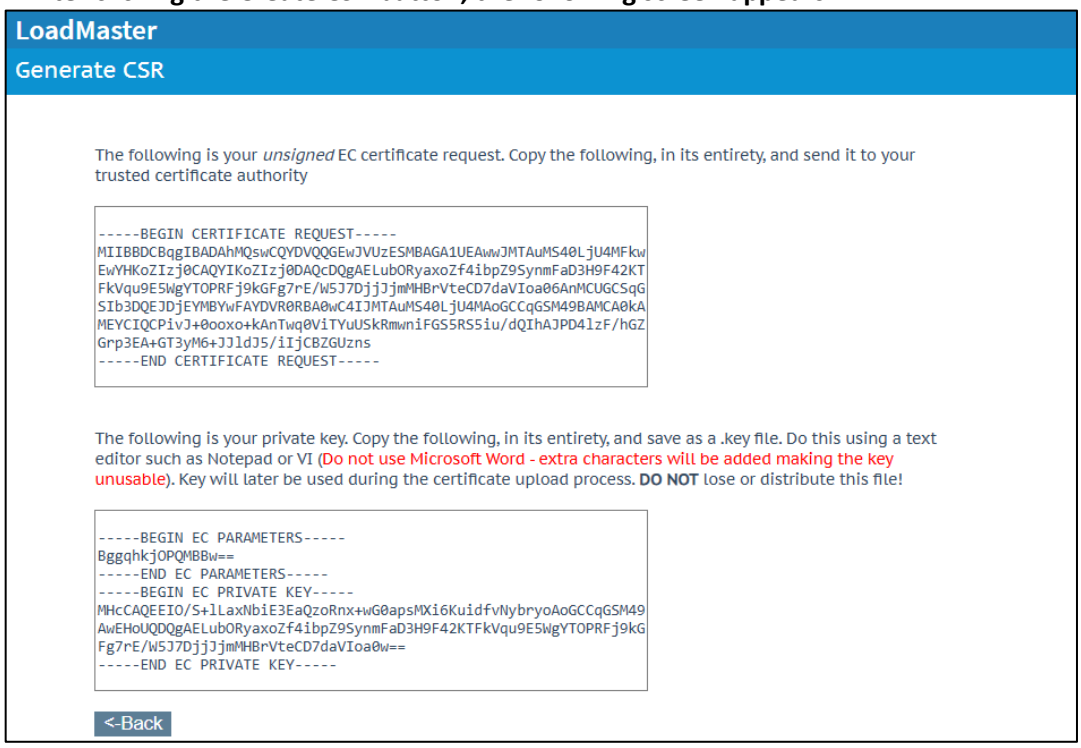
- **Email Address**

  The email address of the responsible person or organization that should be contacted regarding this certificate.

- **SAN/UCC Names**

  A space-separated list of alternate names.

2. **Fill in the details in the resulting screen. The Common Name field is mandatory, all other fields are optional.**
3. **Click Create CSR.**
4. **Alter clicking the Create CSR button, the following screen appears:**



5. **The top part of the screen should be copied and pasted into a plain text file and sent to the Certificate Authority of your choice. They will validate the information and return a validated certificate.**
6. **The lower part of the screen is your private key and should be kept in a safe place. This key should not be disseminated as you will need it to use the certificate. Copy and paste the private key into a plain text file (do not use an application such as Microsoft Word) and keep the file safe.**
7. **The TSF shall validate the chain of certificates from the Root CA upon receiving the CA Certificate Response.**

The relevant information is found in the following section(s): **X509 CERTIFICATE**

Upon investigation, the evaluator found that the AGD states that: **The TOE performs X.509v3 certificate validation for all TLS connections. If certificate verification fails for any reason, the connection attempt fails, and the trusted channel is not established. There are no fallback authentication mechanisms for failed certificate authentication. The TOE provides a trust store to manage and store X.509v3 certificates used for authentication and trusted connections.**

Verdict:

PASS.

5.3.2.3 FIA_X509_EXT.3 X509 CERTIFICATE REQUESTS

5.3.2.3.1 FIA_X509_EXT.3 TSS

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

Evaluator Findings:
If the ST author selects "device-specific information", the evaluator verified that the TSS contains a description of the device-specific fields used in certificate requests. The relevant information is found in the following section(s): TOE Summary Specification **FIA_X509_EXT.2 X.509 CERTIFICATE AUTHENTICATION (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF is capable of generating certificate signing request that contain the public key, device specific information (e.g. email address, requested SAN name), Common Name, Organization, Organizational Unit, Country.**

Verdict:

PASS.

5.3.2.3.2 FIA_X509_EXT.3 AGD

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

Evaluator Findings:
The evaluator checked and ensured that the AGD contains instructions on requesting certificates from a CA, including generation of a Certificate Request. If the ST author selects "Common Name", "Organization", "Organizational Unit", or "Country", the evaluator ensured that the AGD includes instructions for establishing these fields before creating the Certification Request. The relevant information is found in the following section(s): **GENERATE CSR (CERTIFICATE SIGNING REQUEST)** Upon investigation, the evaluator found that the AGD states that: **The TOE is capable of generating a Certificate Signing Request (CSR) that includes the public key along with device-specific information such as email address, requested Subject Alternative Name (SAN), Common Name (CN), Organization (O), Organizational Unit (OU), and Country (C). The following instructions explain how to establish these fields before creating the CSR.** • **Complete the Certificate Signing Request (CSR) form and click the Create CSR button.** • **If Self-Signed Certificate Handling is set to EC certs with an EC signature (in Certificates & Security > Remote Access), CSR generation is restricted to the administrative (bal) user only. If Self-Signed Certificate Handling is set to a different value, all users (regardless of their permissions) can generate CSRs.**

1. **In the main menu, go to Certificates & Security > Generate CSR.**

The following are the instructions for establishing the above fields in the CSR:

- **Letter Country Code (ex. US)**

  The 2-letter country code that should be included in the certificate, for example US should be entered for the United States.

- **State/Province (Entire Name – New York, not NY)**

  The state which should be included in the certificate. Enter the full name here, for example New York, not NY.

- **City**

  The name of the city that should be included in the certificate.

- **Company (Organization)**

  The name of the organization or company which should be included in the certificate.

- **Organization Unit (e.g., Marketing, Finance, Sales)**

  The department or organizational unit that should be included in the certificate.

- **Common Name**

  The Fully Qualified Domain Name (FQDN) for your web server.

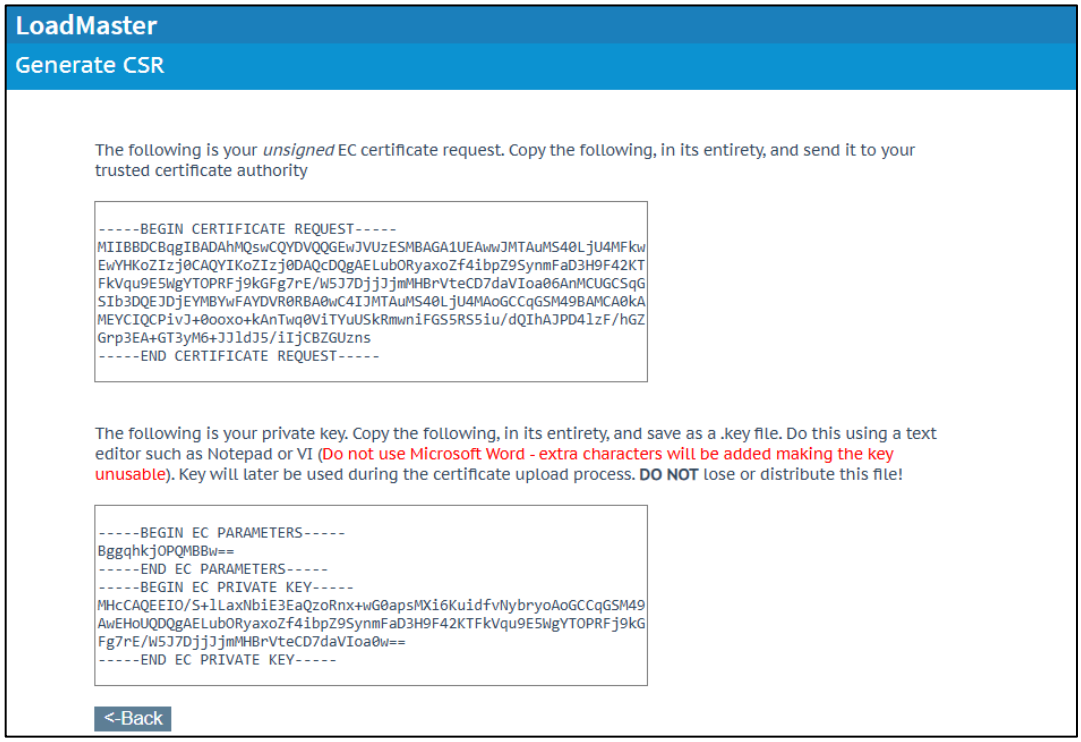
- **Email Address**

  The email address of the responsible person or organization that should be contacted regarding this certificate.

- **SAN/UCC Names**

  A space-separated list of alternate names.

2. **Fill in the details in the resulting screen. The Common Name field is mandatory, all other fields are optional.**
3. **Click Create CSR.**
4. **Alter clicking the Create CSR button, the following screen appears:**



5. **The top part of the screen should be copied and pasted into a plain text file and sent to the Certificate Authority of your choice. They will validate the information and return a validated certificate.**
6. **The lower part of the screen is your private key and should be kept in a safe place. This key should not be disseminated as you will need it to use the certificate. Copy and paste the private key into a plain text file (do not use an application such as Microsoft Word) and keep the file safe.**
7. **The TSF shall validate the chain of certificates from the Root CA upon receiving the CA Certificate Response.**

Verdict:

PASS.

5.3.2.4 FIA_AFL.1 AUTHENTICATION FAILURE MANAGEMENT

5.3.2.4.1 FIA_AFL.1 TSS

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

Evaluator Findings:
The evaluator shall examine the TSS to determine that it contains a description, for each supported method for remote administrative actions, of how successive unsuccessful authentication attempts are detected and tracked. The TSS shall also describe the method by which the remote administrator is prevented from successfully logging on to the TOE, and the actions necessary to restore this ability. The relevant information is found in the following section(s): TOE Summary Specification **FIA_AFL.1 AUTHENTICATION FAILURE HANDLING (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that **The TSF supports TLS-based remote administration. The TSF blocks remote authentication attempts after a configurable number of failed attempts for password -based authentication mechanism. The security administrator can configure the threshold to be from 1 through 999. Once an account is locked, the account must be unlocked using the "bal" account through the WUI.**

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 shall examine the TSS to confirm that the TOE ensures that authentication failures by remote administrators cannot lead to a situation where no administrator access is available, either permanently or temporarily (e.g. by providing local logon which is not subject to blocking). The relevant information is found in the following section(s): TOE Summary Specification **FIA_AFL.1 AUTHENTICATION FAILURE HANDLING (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **Once an account is locked, the account must be unlocked using the "bal" account through the WUI. The administration of the TSF is always possible, because the TSF never locks the local console.**

Verdict:

PASS.

5.3.2.4.2 FIA_AFL.1 AGD

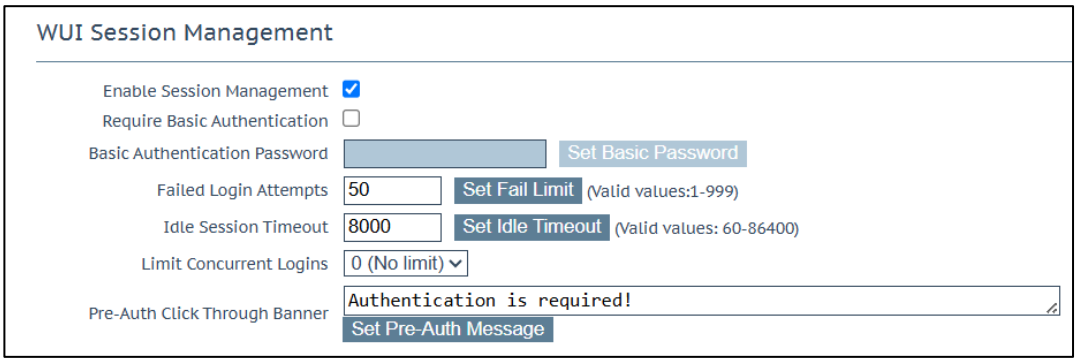
The evaluator shall examine the guidance documentation to ensure that instructions for configuring the number of successive unsuccessful authentication attempts and time period (if implemented) are provided, and that the process of allowing the remote administrator to once again successfully log on is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

Evaluator Findings:

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

The relevant information is found in the following section(s): **FAILED LOGIN ATTEMPTS**

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



The TSF blocks remote authentication attempts after a configurable number of consecutive failed login attempts. This policy is enforced for remote administrative access via the WUI (TLS-based).

To configure the maximum number of failed login attempts:

1. **In the left frame menu of the Web User Interface (WUI), navigate to:**

   System Configuration > Certificate & Security > Admin WUI Access > Failed Login Attempts

2. **Set the desired value in the Failed Login Attempts field.**
   - **This specifies the number of times a user can fail to authenticate before their account is blocked.**
   - **Valid values range from 1 to 999.**
3. **If a user account becomes blocked due to failed login attempts:**
   - **The account must be manually unblocked by the bal user or another user with All Permissions set.**
   - **If the bal user is blocked, there is an enforced cool-down period of 10 minutes, after which the bal user can log in again.**

The TSF ensures that administration is always possible, even in the event of a lockout. Local console access is never locked, ensuring that a security administrator can always regain access to the system even if all remote interfaces are temporarily inaccessible.

This lockout policy applies to all remote authentication channels, for access over TLS (WUI).

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

Evaluator Findings:
The evaluator examined the guidance documentation to confirm that it describes, and identifies the importance of, any actions that are required in order to ensure that administrator access will always be maintained, even if remote administration is made permanently or temporarily unavailable due to blocking of accounts as a result of FIA_AFL.1. The relevant information is found in the following section(s): **FAILED LOGIN ATTEMPTS** Upon investigation, the evaluator found that the AGD states that: **The TSF ensures that administration is always possible, even in the event of a lockout. Local console access is never locked, ensuring that a security administrator can always regain access to the system even if all remote interfaces are temporarily inaccessible.**

Verdict:

PASS.

5.3.2.5 FIA_UAU.7 PROTECTED AUTHENTICATION FEEDBACK

5.3.2.5.1 FIA_UAU.7 TSS

None.

5.3.2.5.2 FIA_UAU.7 AGD

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

Evaluator Findings:
The evaluator examined the guidance documentation to determine that any necessary preparatory steps to ensure authentication data is not revealed while entering for each local login allowed. The relevant information is found in the following section(s): **AUTHENTICATION AND ROLE-BASED ACCESS CONTROL** Upon investigation, the evaluator found that the AGD states that: **No administrative or security-relevant functions are available until the user has successfully authenticated, and the feedback during authentication is obscured. Only authenticated users with the appropriate permissions can view or modify TSF data. Unauthorized users cannot perform any security-related actions.**

Verdict:

PASS.

5.3.2.6 FIA_PMG_EXT.1 PASSWORD MANAGEMENT

5.3.2.6.1 FIA_PMG_EXT.1 TSS

The evaluator shall check that the TSS:

a. lists the supported special character(s) for the composition of administrator passwords.

b. to ensure that the minimum_password_length parameter is configurable by a Security Administrator.

c. 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 checked that the TSS: a. lists the supported special character(s) for the composition of administrator passwords. b. Ensured that the minimum_password_length parameter is configurable by a Security Administrator. c. lists the range of values supported for the minimum_password_length parameter. The listed range shall include the value of 15. The relevant information is found in the following section(s): TOE Summary Specification **FIA_PMG_EXT.1 PASSWORD MANAGEMENT (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF allows administrators passwords to be composed of any printable ASCII character (i.e. 0x20-0x7E inclusive).** **Supported special characters are as follows:** • **“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, “)”, “ “, “‘”, “‘”, “+”, “,”, “-“, “.”, “/”, “:”, “;”, “<”, “=”, “>”, “?”, “@”, “[“, “\”, “]”, “_”, “`”, “{“, “|”, “}”, “~”;** **The TSF allows the security administrator to configure the minimum password length to be 8-16 characters.**

Verdict:

PASS.

5.3.2.6.2 FIA_PMG_EXT.1 AGD

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

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

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

Evaluator Findings:
The evaluator examined the guidance documentation to determine that it: a. identifies the characters that may be used in passwords and provides guidance to security administrators on the composition of strong passwords, and b. provides instructions on setting the minimum password length and describes the valid minimum password lengths supported. The relevant information is found in the following section(s): **SET MINIMUM PASSWORD LENGTH** Upon investigation, the evaluator found that the AGD states that: **1. In the left frame menu, click System Configuration > System Administration > User Management to set the desired Minimum Password Length (default is 8).** **2. The TSF allows administrators passwords to be composed of any combination of upper and lower case letters, numbers and the following printable ASCII character (i.e. 0x20-0x7E inclusive):** **3. “!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, “)”, “ “, “‘”, “‘”, “+”, “,”, “-“, “.”, “/”, “:”, “;”, “<”, “=”, “>”, “?”, “@”, “[“, “\”, “]”, “_”, “`”, “{“, “|”, “}”, “~”;** **4. The Minimum Password Length should be configured between 8 and 16 characters.**

Verdict:

PASS.

5.3.3 PROTECTION OF THE TSF (FPT)

5.3.3.1 FPT_APW_EXT.1 PROTECTION OF ADMINISTRATOR PASSWORDS

5.3.3.1.1 FPT_APW_EXT.1 TSS

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

Evaluator Findings:
The evaluator examined the TSS to determine that it details all authentication data that are subject to this requirement, and the method used to obscure the plaintext password data when stored. The TSS also detailed passwords are stored in such a way that they are unable to be viewed through an interface designed specifically for that purpose, as outlined in the application note. The relevant information is found in the following section(s): TOE Summary Specification **FPT_APW_EXT.1 PROTECTION OF ADMINISTRATOR PASSWORDS (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF supports a 'bal' account (superuser) and administrator accounts. Both account types belong to the Security Administrator role. The TSF stores administrative passwords protected by a SHA-512 hash. Passwords stored in a secure location which is not accessible to users. Secure hash functions ensure that it's computationally impossible to recover a plaintext from its hashed value.**

Verdict:

PASS.

5.3.3.1.2 FPT_APW_EXT.1 AGD

None.

5.3.4 SECURITY MANAGEMENT (FMT)

5.3.4.1 FMT_MOF.1/SERVICES MANAGEMENT OF SECURITY FUNCTIONS BEHAVIOUR

5.3.4.1.1 FMT_MOF.1/SERVICES TSS

For distributed TOEs see Section 2.4.1.1.

Evaluator Findings:
NA, This SFR is not claimed in the ST.

For non-distributed TOEs, the evaluator shall ensure the TSS lists the services the Security Administrator is able to start and stop and how that operation is performed.

Evaluator Findings:
The evaluator examined the TSS and ensured that, for non-distributed TOEs, it lists the services the Security Administrator is able to start and stop and how that operation is performed. The relevant information is found in the following section(s): TOE Summary Specifications **FMT_MOF.1/SERVICES MANAGEMENT OF SECURITY FUNCTIONS BEHAVIOUR (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF allows the security administrator to start and stop following services.** • **WUI Service** • **syslog services**

For WUI Service, **The WUI service can be stopped using either the WUI itself or from the Console.** **For syslog Service,** **it can be start and stop using WUI only.**

Verdict:

PASS.

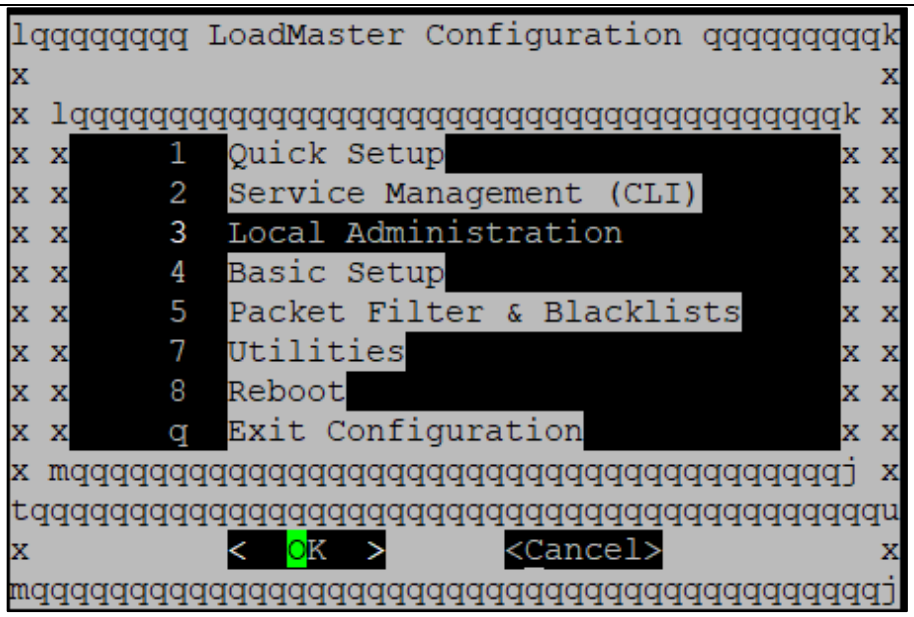
5.3.4.1.2 FMT_MOF.1/SERVICES AGD

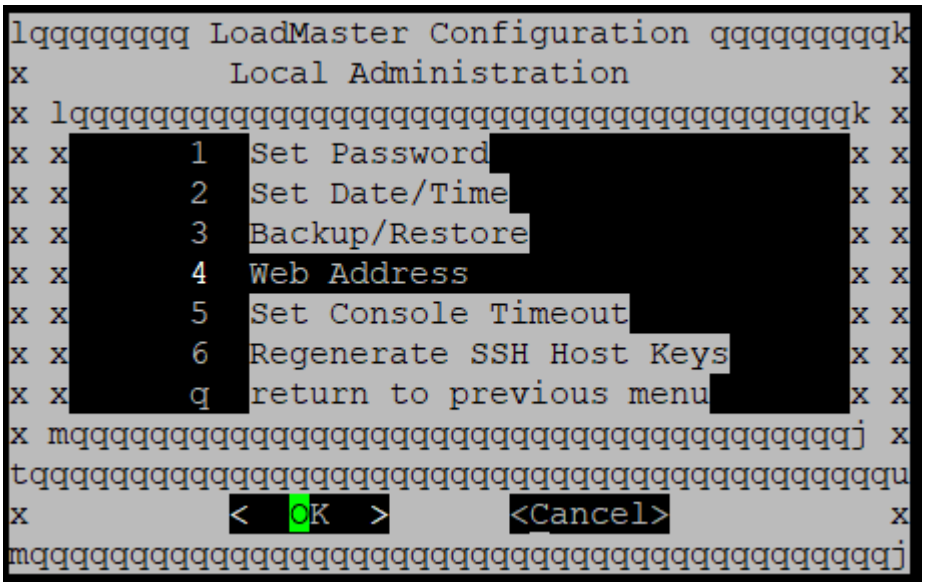
For distributed TOEs see Section 2.4.1.2.

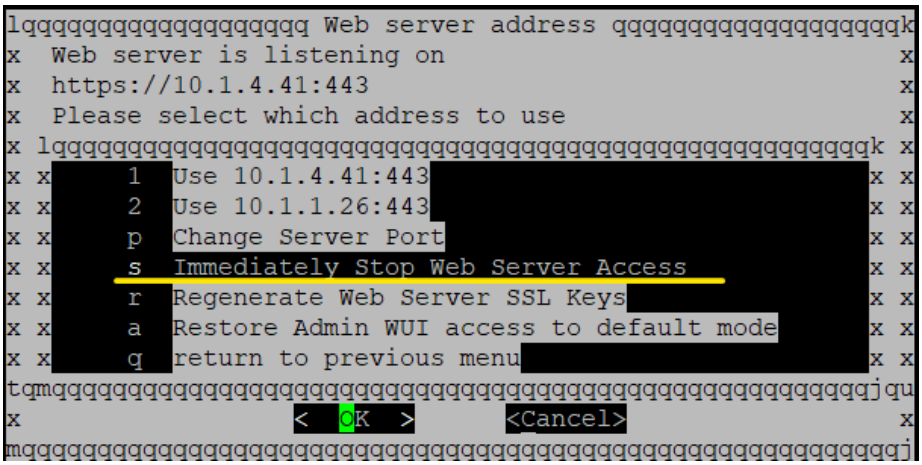
Evaluator Findings:
NA, This SFR is not claimed in the ST.

For non-distributed TOEs, the evaluator shall also ensure the Guidance Documentation describes how the TSS lists the services the Security Administrator is able to start and stop and how that operation is performed.

Evaluator Findings:
The evaluator examined the AGD and ensured that, for non-distributed TOEs, it describes how the TSS lists the services the Security Administrator is able to start and stop and how that operation is performed. The relevant information is found in the following section(s): **START/STOP WEBUI ACCESS** Upon investigation, the evaluator found that the AGD states that: **The TOE allows the security administrator to control access to the Web User Interface (WUI) either directly from the local console or partially through the WUI itself.** **To perform this action via local console:** **1. Access the TOE via local console.** **2. Navigate to Local Administration > Web Address > Immediately Stop/Start Web Server Access.** **3. Use this option to:** **○ Stop: Immediately disable access to the WUI, blocking all remote browser-based administrative access.** **○ Start: Re-enable access to the WUI over TLS, allowing remote administration.** **The screenshot below shows the exact location of this option within the local console interface.**

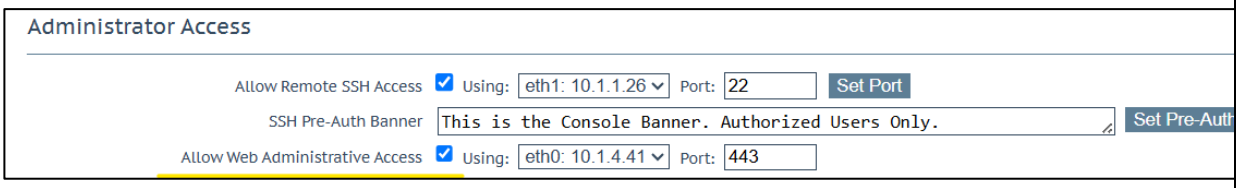

LoadMaster Configuration
1 Quick Setup
2 Service Management (CLI)
3 Local Administration
4 Basic Setup
5 Packet Filter & Blacklists
7 Utilities
8 Reboot
q Exit Configuration
< OK >
<Cancel>


LoadMaster Configuration
Local Administration
1 Set Password
2 Set Date/Time
3 Backup/Restore
4 Web Address
5 Set Console Timeout
6 Regenerate SSH Host Keys
q return to previous menu
< OK >
<Cancel>


Web server address
Web server is listening on
https://10.1.4.41:443
Please select which address to use
1 Use 10.1.4.41:443
2 Use 10.1.1.26:443
p Change Server Port
s Immediately Stop Web Server Access
r Regenerate Web Server SSL Keys
a Restore Admin WUI access to default mode
q return to previous menu
< OK >
<Cancel>

Alternatively, to disable WUI access via WUI:

1. **Navigate to Certificates & Security > Remote Access > Administrator Access.**
2. **Uncheck the Allow Web Administrative Access checkbox.**



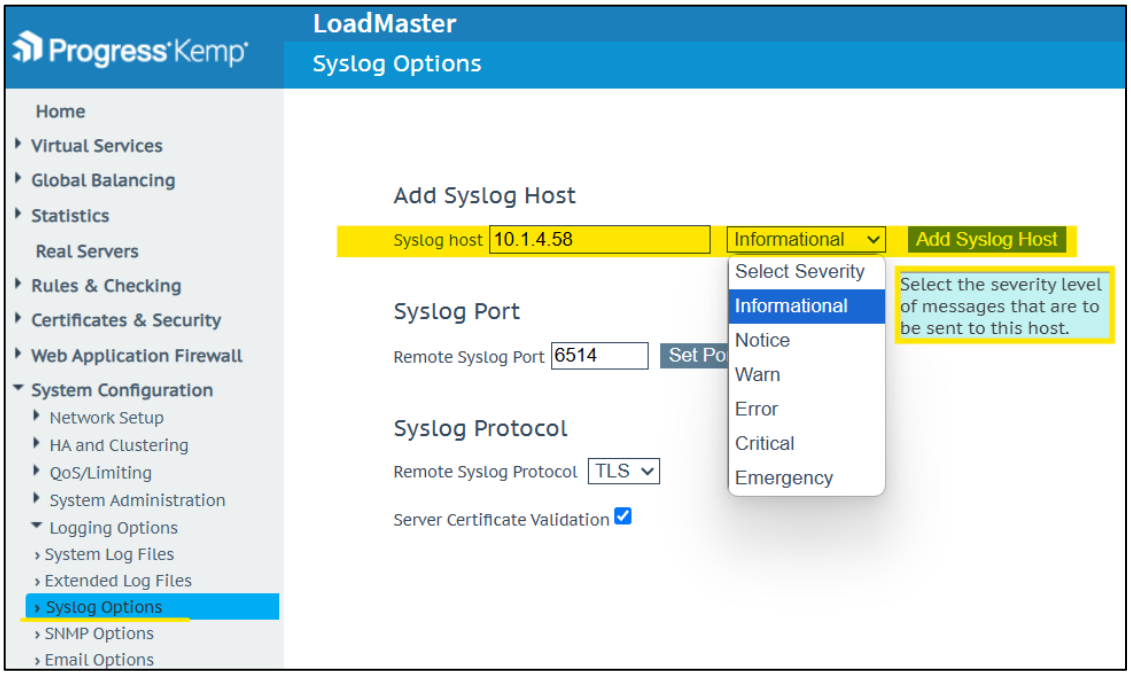
Note: Through WUI, you can only disable WUI access. Once disabled, WUI access can only be re-enabled via the local console using the steps outlined above.

This functionality enables the administrator to manage the availability of the WUI as needed for security or maintenance purposes.

The relevant information is found in the following section(s): **STARTING THE SYSLOG SERVICE**

Upon investigation, the evaluator found that the AGD states that: **To start or enable the syslog service:**

1. **Navigate to:**
   System Administration > Logging Options > Syslog Options
2. **Under the Syslog Hosts section, locate the Syslog Level dropdown menu.**
3. **From the dropdown, select any level except None (e.g., Alert, Critical, Error, etc.).**

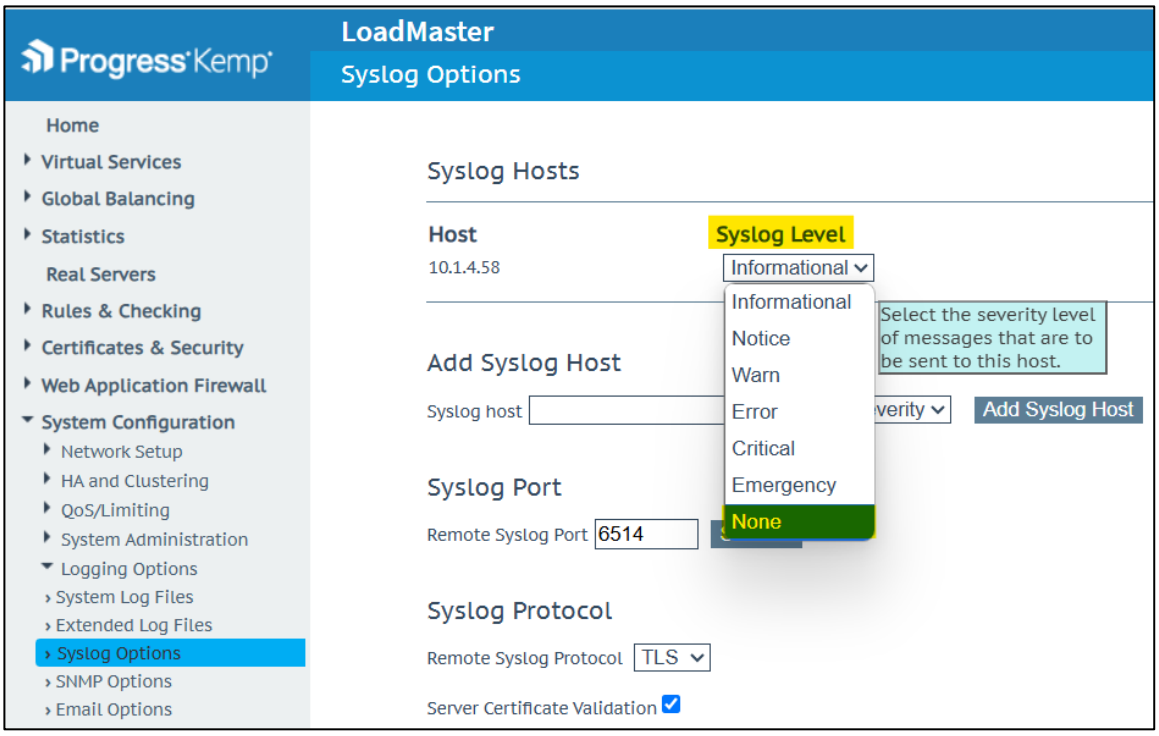


Selecting any valid syslog level will start the syslog service and enable log forwarding to the configured syslog hosts.

The relevant information is found in the following section(s): **STOPPING THE SYSLOG SERVICE**

Upon investigation, the evaluator found that the AGD states that: **To stop or disable the syslog service:**

1. **Navigate to:**
   System Administration > Logging Options > Syslog Options
2. **Under the Syslog Hosts section, locate the Syslog Level dropdown menu.**
3. **From the dropdown, select None.**



This will stop the syslog service and prevent the TOE from sending logs to the configured syslog servers.

Verdict:

PASS.

5.3.4.2 FMT_MOF.1/FUNCTIONS MANAGEMENT OF SECURITY FUNCTIONS BEHAVIOUR

5.3.4.2.1 FMT_MOF.1/FUNCTIONS TSS

For distributed TOEs see Section 2.4.1.1.

Evaluator Findings:
NA, This SFR is not claimed in the ST.

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 TSS and ensured that, for non-distributed TOEs, it 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). The relevant information is found in the following section(s): TOE Summary Specification - **FMT_MOF.1/Functions** Upon investigation, the evaluator found that the TSS states that: **The TOE restricts the ability to determine and modify the behavior of transmission of audit data to an external IT entity to Security Administrators.** **The WUI contains all functions for configuring the transmission of the audit data. The WUI, and therefore the functions, are only available to successfully authenticated Security Administrators.**

Verdict:

PASS.

5.3.4.2.2 FMT_MOF.1/FUNCTIONS AGD

For distributed TOEs see Section 2.4.1.2.

Evaluator Findings:
NA, This SFR is not claimed in the ST.

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 Guidance Documentation and ensured that, for non-distributed TOEs, it 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. The relevant information is found in the following section(s): The relevant information is found in the following section(s): **Secure Remote Logging** Upon investigation, the evaluator found that the AGD states that: **The LoadMaster can produce various warning and error messages using the syslog protocol. These messages are normally stored locally. Syslog messages are transmitted securely using TLS to remote servers. The LoadMaster uses**

OCSP to check the validity of the server certificates supplied by configured syslog servers. If these checks fail, connections to the server are not permitted.

1. In the left frame menu, click System Configuration > System Log Files > Syslog Options.

2. By entering the relevant IP address in the Syslog host text box and select the severity and click Add Syslog Host.

3. Also enter the Syslog Port enter any port other than 601 and click on Set Port.

4. Select the Syslog Protocol either TCP, UDP or TLS.

5. Enable the Server Certificate Validation.

Note: secure syslog channel is restricted to TLSv1.2

NOTE: If the TLS connection to the syslog server is unintentionally broken, the administrator should verify the connectivity to the syslog server. Once connectivity is restored, the administrator must manually re-initiate the syslog connection.

Verdict:

PASS.

5.3.4.3 FMT_MTD.1/CRYPTOKEYS MANAGEMENT OF TSF DATA

5.3.4.3.1 FMT_MTD.1/CRYPTOKEYS TSS

For distributed TOEs see Section 2.4.1.1.

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

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

Evaluator Findings:
The evaluator examined the TSS and ensured that, for non-distributed TOEs, it lists the keys the Security Administrator is able to manage to include the options available (e.g. generating keys, importing keys, modifying keys or deleting keys) and how that how those operations are performed. The relevant information is found in the following section(s): TOE Summary Specification **FMT_MTD.1/CRYPTOKEYS MANAGEMENT OF TSF DATA (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF allows the security administrator to generate cryptographic keys associated with the TSF's self-signed web server certificate or Certificate Signing Requests. TSF also allows users with Security Administrator privileges to import and delete certificates from the trust store.**

Verdict:

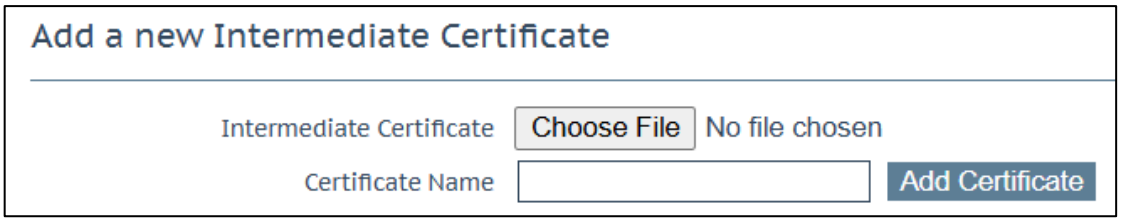
PASS.

5.3.4.3.2 FMT_MTD.1/CRYPTOKEYS AGD

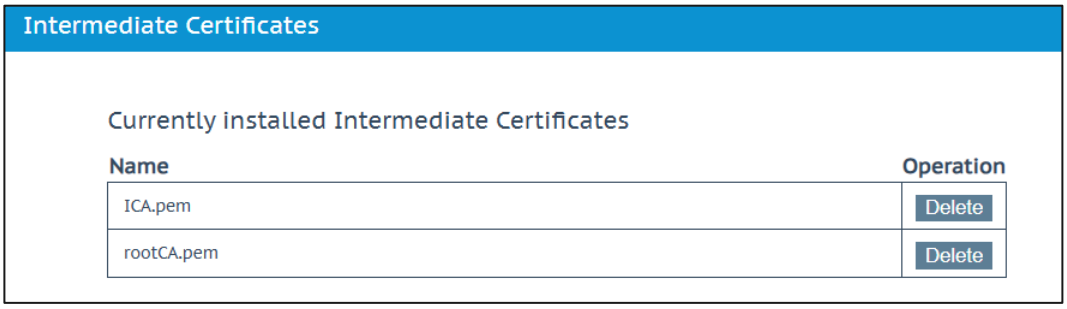
For distributed TOEs see Section 2.4.1.2.

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

For non-distributed TOEs, the evaluator shall also ensure the Guidance Documentation describes how the operations are performed on the keys the Security Administrator is able to manage.

Evaluator Findings:
The evaluator ensured the Guidance Documentation describes how the operations are performed on the keys the Security Administrator is able to manage. The relevant information is found in the following section(s): **INTERMEDIATE CERTIFICATE** Upon investigation, the evaluator found that the AGD states that: **The TSF allows the security administrator to generate cryptographic keys associated with TSF's self-signed web server certificate or Certificate Signing Requests. The Intermediate Certificates field allows you to assign intermediate and root certificates. If you already have a certificate, or you have received one from a CSR, you can install the certificate by clicking the Choose File button. Navigate to and select the certificate and then enter the desired Certificate Name. The name can only contain alpha characters with a maximum of 32 characters.** **Uploading several consecutive intermediate certificates within a single piece of text, as practiced by some certificate vendors such as GoDaddy, is allowed. The uploaded file is split into the individual certificates.** **You can also delete the certificates by clicking the 'Delete' button shown in the first screenshot of this section.** **If a connection is not possible because the validity of a certificate cannot be determined, there is no override option. A valid certificate must be presented. This may include installing required certificates in the trust store, i.e. uploading the certificates in the Intermediate Certificate field.** **To upload Root and Intermediate CA Certificates, before enabling certificate-based admin WUI access:** 1. **Navigate to Certificates & Security → Intermediate Certificates.** 2. **Upload the issuing CA and Root CA certificates required to validate administrator certificates for client authentication.**    **Note: Ensure that the server certificate is an X.509 certificate with an ECDSA public key using the secp256r1 (P-256) curve, as required by the evaluated configuration.** **TSF allows the security administrator to generate cryptographic keys associated with TSF's self-signed web server certificate or Certificate Signing Requests.**

The same screen as shown below displays the list of installed certificates along with their assigned names.



The relevant information is found in the following section(s): **GENERATE CSR (CERTIFICATE SIGNING REQUEST)**

Upon investigation, the evaluator found that the AGD states that: **The TOE is capable of generating a Certificate Signing Request (CSR) that includes the public key along with device-specific information such as email address, requested Subject Alternative Name (SAN), Common Name (CN), Organization (O), Organizational Unit (OU), and Country (C). The following instructions explain how to establish these fields before creating the CSR.**

- **Complete the Certificate Signing Request (CSR) form and click the Create CSR button.**
- **If Self-Signed Certificate Handling is set to EC certs with an EC signature (in Certificates & Security > Remote Access), CSR generation is restricted to the administrative (bal) user only. If Self-Signed Certificate Handling is set to a different value, all users (regardless of their permissions) can generate CSRs.**

1. **In the main menu, go to Certificates & Security > Generate CSR.**

The following are the instructions for establishing the above fields in the CSR:

- **Letter Country Code (ex. US)**

  The 2-letter country code that should be included in the certificate, for example US should be entered for the United States.

- **State/Province (Entire Name – New York, not NY)**

  The state which should be included in the certificate. Enter the full name here, for example New York, not NY.

- **City**

  The name of the city that should be included in the certificate.

- **Company (Organization)**

  The name of the organization or company which should be included in the certificate.

- **Organization Unit (e.g., Marketing, Finance, Sales)**

  The department or organizational unit that should be included in the certificate.

- **Common Name**

  The Fully Qualified Domain Name (FQDN) for your web server.

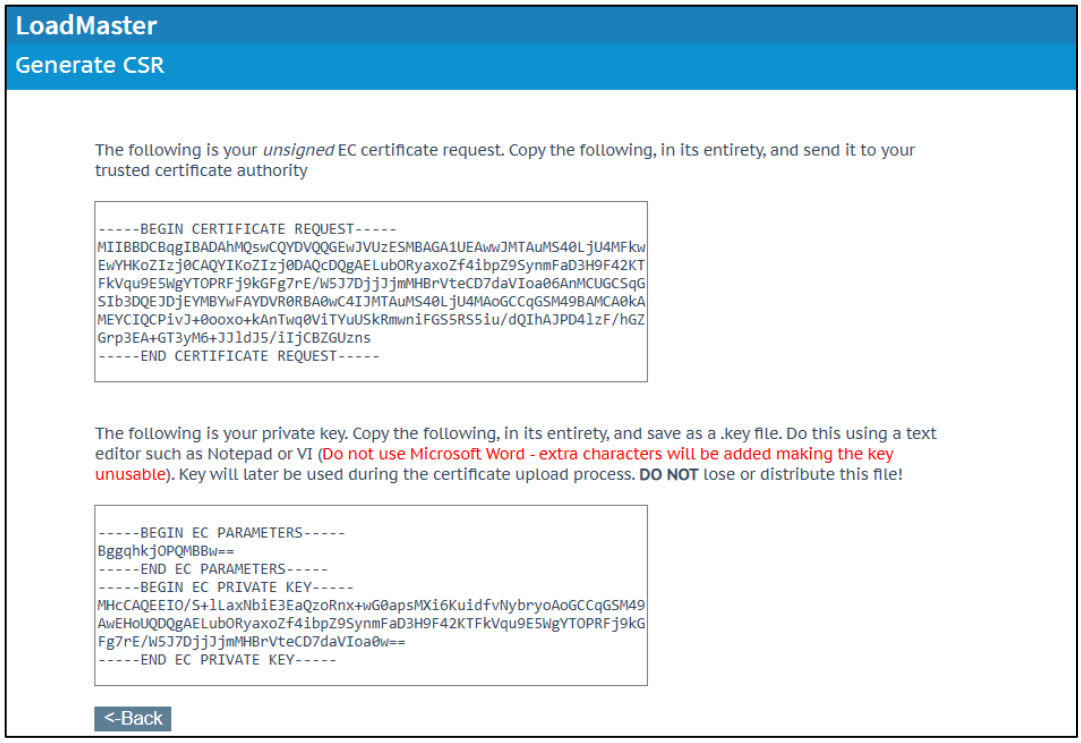
- **Email Address**

  The email address of the responsible person or organization that should be contacted regarding this certificate.

- **SAN/UCC Names**

  A space-separated list of alternate names.

2. **Fill in the details in the resulting screen. The Common Name field is mandatory, all other fields are optional.**
3. **Click Create CSR.**
4. **Alter clicking the Create CSR button, the following screen appears:**



5. **The top part of the screen should be copied and pasted into a plain text file and sent to the Certificate Authority of your choice. They will validate the information and return a validated certificate.**
6. **The lower part of the screen is your private key and should be kept in a safe place. This key should not be disseminated as you will need it to use the certificate. Copy and paste the private key into a plain text file (do not use an application such as Microsoft Word) and keep the file safe.**
7. **The TSF shall validate the chain of certificates from the Root CA upon receiving the CA Certificate Response.**

Verdict:

PASS.

5.3.5 TOE ACCESS (FTA)

5.3.5.1 FTA_SSL_EXT.1 TSF-INITIATED SESSION LOCKING

5.3.5.1.1 FTA_SSL_EXT.1 TSS

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

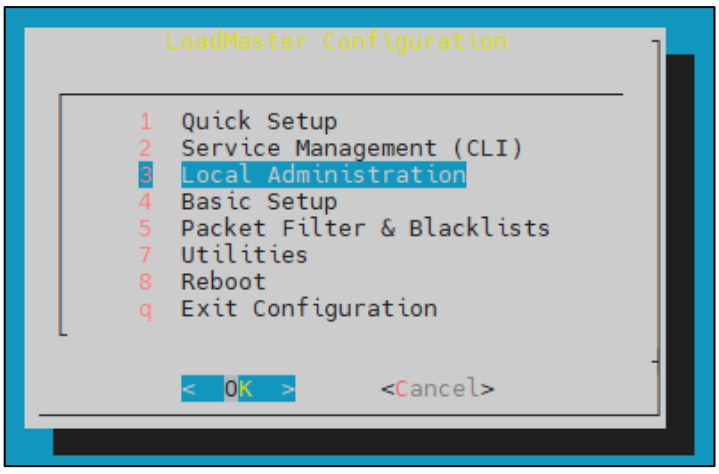
Evaluator Findings:
The evaluator examined the TSS to determine that it details whether local administrative session locking or termination is supported and the related inactivity time period settings. The relevant information is found in the following section(s): TOE Summary Specification **FTA_SSL_EXT.1 TSF-INITIATED SESSION LOCKING (CPP_ND_V3.0E)** Upon investigation, the evaluator found that the TSS states that: **The TSF supports local administrative session termination. The TSF terminates local sessions after 1-1440 minutes of inactivity. A configured inactivity period will be applied to local sessions. When the local session has been idle for more than the configured period, the session will be terminated and will require authentication to establish a new session.**

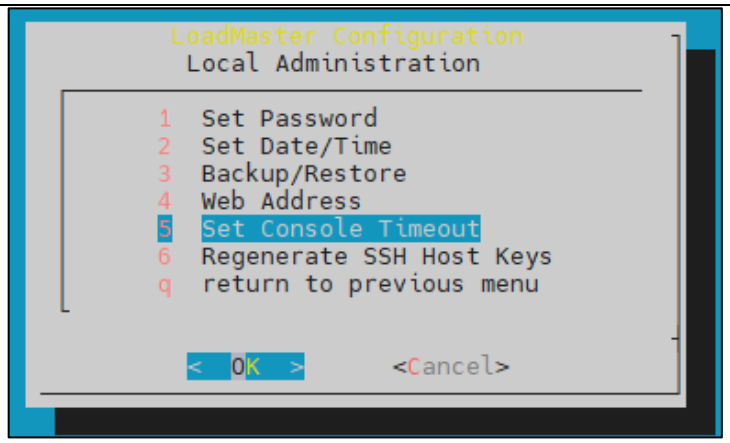
Verdict:

PASS.

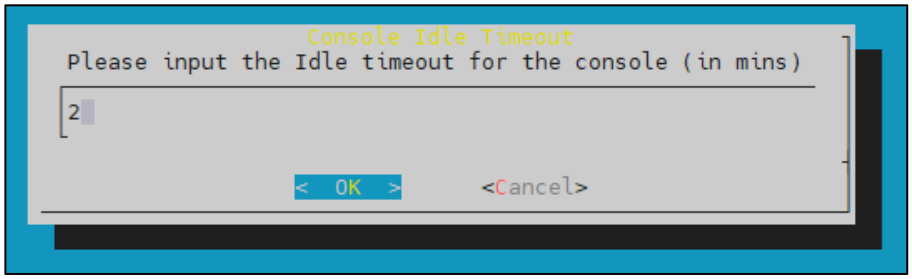
5.3.5.1.2 FTA_SSL_EXT.1 AGD

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

Evaluator Findings:
The evaluator confirmed that the guidance documentation states whether local administrative session locking or termination is supported and instructions for configuring the inactivity time period The relevant information is found in the following section(s): **IDLE SESSION TIMEOUT** Upon investigation, the evaluator found that the TSS states that: Upon investigation, the evaluator found that the AGD states that: **To configure the idle timeout setting via console:** 1. **After logging in to the CLI, Navigate to:** **Local Administration > Set Console Timeout**   



2. **When prompted, input the desired idle timeout value.**
    - **Valid values range from 1 minute (60 seconds) to 1440 minutes (24 hours).**



    - **This setting determines how long a CLI/console session can remain idle before it is automatically terminated by the TSF.**

Verdict:

PASS.

6 SECURITY ASSURANCE REQUIREMENTS

6.1 ASE: SECURITY TARGET EVALUATION

6.1.1 GENERAL EVALUATION ACTIVITY

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

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

Verdict:
PASS.

6.2 ADV: DEVELOPMENT

6.2.1 EVALUATION ACTIVITY

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

In this context, TSFI are deemed security relevant if they are used by the administrator to configure the TOE, or to perform other administrative functions (e.g. audit review or performing updates). Explicitly labeling TSFI as security relevant or non-security relevant is not necessary. A TSFI is implicitly security relevant if it is used to satisfy an evaluation activity, or if it is identified in the ST or guidance documentation as adhering to the security policies (as presented in the SFRs). The intent is that these interfaces will be adequately tested and having an understanding of how these interfaces are used in the TOE is necessary to ensure proper test coverage is applied. According to the description above 'security relevant' corresponds to the combination of 'SFRenforcing' and 'SFR-supporting' as defined in CC Part 3, paragraph 224 and 225.

The set of TSFI that are provided as evaluation evidence are contained in the Security Target and the guidance documentation.

Evaluator Findings:
TOE Design information that can be made public is available in the guidance documentation and in the ST. Any sensitive or proprietory information required by this protection profile is not to be made public. It is not necessary to provide a complete specification of the TSFIs. For NDcPP, additional “functional specification” documentation is not necessary because this requirement is satisfied by multiple other documents (AGD, TSS, and Testing). All associated activities are covered in the Test Report, ST, and AGD documents. NDcPP3.0e, section 7.2.1 states that: “For this cPP, the Evaluation Activities for this family focus on understanding the interfaces presented in the TSS in response to the functional requirements and the interfaces presented in the AGD documentation.” All of the above information is applicable to the ADV Evaluation Activities (5.2.1.1, 5.2.1.2, and 5.2.1.3) in NDcPP3.0e-SD. The evaluator examined the ST (Security Target) and the AGD (interface documentation) to verify that it describes the purpose and method of use for each TSFI that is identified as being security relevant.

The evaluator examined the entire AGD. The evaluator verified the AGD describes the purpose and method of use for each security relevant TSFI by verifying the AGD satisfies all the AGD Evaluation Activities. During testing, the evaluator used the product and its interfaces extensively and did not find any areas that were deficient.

Verdict:
PASS.

6.2.2 EVALUATION ACTIVITY

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

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

Verdict:
PASS.

6.2.3 EVALUATION ACTIVITY

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

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

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

The required EAs define the design and interface information required to meet ADV_FSP.1. If the evaluator is unable to perform some EA, then the evaluator shall conclude that an adequate functional specification has not been provided.

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

Verdict:
PASS.

6.3 AGD: GUIDANCE DOCUMENTATION

6.3.1 OPERATIONAL USER GUIDANCE (AGD_OPE.1)

6.3.1.1 EVALUATION ACTIVITY

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

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

Verdict:
PASS.

6.3.1.2 EVALUATION ACTIVITY

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

Evaluator Findings:
The evaluator examined the section titled **Supported Platforms** of the AGD and it was used to determine the verdict of this assurance activity. The ST claims LoadMaster X25-NG, LoadMaster X40-NG ,ECS CM H2 NG, ECS CM H3 NG and virtual LoadMaster platforms, and the operational guidance documents cover the configuration and use of these platforms. An additional VMware platform installation document is available from the Progress website for Virtual LoadMaster installation prerequisites and steps. Based on these findings, this assurance activity is considered satisfied.

Verdict:
PASS.

6.3.1.3 EVALUATION ACTIVITY

The evaluator shall ensure that the Operational guidance contains instructions for configuring any cryptographic implementation associated with the evaluated configuration of the TOE. It shall provide a

warning to the administrator that use of other cryptographic implementations was not evaluated nor tested during the CC evaluation of the TOE.

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

Verdict:
PASS.

6.3.1.4 EVALUATION ACTIVITY

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

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

Verdict:
PASS.

6.3.1.5 EVALUATION ACTIVITY

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

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

b. The evaluator shall verify that this process includes 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).

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 sections titled **CC CONFIGURATION PROCESS and DEFAULT CONFIGURATION**. The evaluator verified the guidance documentation describes the process for verifying updates in section titled **INSTALLING AN UPDATE IMAGE** . The evaluator verified the guidance documentation makes it clear which security functionality is covered by the **OPERATIONAL ENVIRONMENT** .

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

Verdict:
PASS.

6.3.2 PREPARATIVE PROCEDURES (AGD_PRE.1)

6.3.2.1 EVALUATION ACTIVITY

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

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

Evaluator Findings:
The evaluator examined the Preparative procedures to ensure they include a description of how the administrator verifies that the operational environment can fulfil its role to support the security functionality. The evaluator reviewed the sections titled **Operational Environment** of the AGD. The evaluator found that these sections describe how the Operational Environment must meet: • **Management Workstation** • **Syslog Server** • **ESXi Server** • **Authentication Server (LDAP)** • **NTP Server** • **CA Server** • **OCSP Server**

Verdict:
PASS.

6.3.2.2 EVALUATION ACTIVITY

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

Evaluator Findings:
The evaluator examined the Preparative procedures to ensure they include a description of how the administrator verifies that the operational environment can fulfil its role to support the security functionality. The evaluator reviewed the sections titled **Operational Environment** of the AGD. The evaluator found that these sections describe how the Operational Environment must meet:

- **Management Workstation providing local console access to the TOE and a browser to connect to the Web User Interface (WUI) over TLSv1.2.**
- **Syslog server that receives audit logs from the TOE over TLSv1.2.**
- **ESXi v7.0 U3 acting as the hypervisor for Virtual LoadMaster.**
- **Authentication server supporting LDAP over TLSv1.2.**
- **NTP server supporting SHA-1 integrity verification with NTPv4.**
- **OCSP server that receives ocsp requests from TOE over HTTP.**

  CA Server that provides signed certificates over HTTP.

Verdict:
PASS.

6.3.2.3 EVALUATION ACTIVITY

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

Evaluator Findings:

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

- **Configuring Administrative Accounts and Passwords**
- **Configuring Console Connections**
- **Configuring the Remote Syslog Server**
- **Configuring Audit Log Options**
- **Configuring Event Logging**
- **Configuring a Secure Logging Channel**
- **Configuring NTP server**
- **Configuring LDAP Server**
- **Generating CSR**
- **Configuring certificates**
- **Configuring Idle session**

Verdict:
PASS.

6.3.2.4 EVALUATION ACTIVITY

The evaluator shall examine the preparative procedures to ensure they include instructions on how to manage the TSF as a product and as a component of the larger Operational Environment in a manner that allows to preserve integrity of the TSF.

The intent of this requirement is to ensure there exists adequate preparative procedures (guidance in most cases) to put the TSF in a secure state (i.e., evaluated configuration). AGD_PRE.1 lists general requirements, the specific assurance activities implementing it are performed as part of FMT_SMF.1, FMT_MTD.1 and FMT_MOF.1 series of SFRs.

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

Verdict:
PASS.

6.3.2.5 EVALUATION ACTIVITY

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

- include instructions to provide a protected administrative capability; and
- identify TOE passwords that have default values associated with them and mandate that they shall 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 **LOGIN VIA CONSOLE** and **CHANGE THE PASSWORD** were used to determine the verdict of this work unit. The AGD describes changing the default password associated with the bal account. Based on these findings, this assurance activity is considered satisfied.

Verdict:
PASS.

6.4 AVA: VULNERABILITY ASSESSMENT

6.4.1 VULNERABILITY SURVEY (AVA_VAN.1)

6.4.1.1 EVALUATION ACTIVITY (DOCUMENTATION)

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

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

The developer shall provide documentation identifying the list of software and hardware components that compose the TOE. Hardware components should identify compute-capable hardware components,

at a minimum that must include the processor, and where applicable, discrete crypto ASICs, TPMs, etc. used by the TOE. Software components include applications, the operating system and other major components that are independently identifiable and reusable (outside of the TOE), for example a web server, protocol or cryptographic implementations, (independently identifiable and reusable components are not limited to the list provided in the example). This additional documentation is merely a list of the name and version number of the components and will be used by the evaluators in formulating vulnerability hypotheses during their analysis.

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

Verdict:
PASS.

6.4.1.2 EVALUATION ACTIVITY

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

Evaluator Findings:
The evaluator documented their analysis and testing of potential vulnerabilities with respect to this requirement. Public searches were performed against all keywords found within the Security Target and AGD that may be applicable to specific TOE components. This included protocols, TOE software version, and TOE hardware to ensure sufficient coverage under AVA. The evaluator searched the Internet for potential vulnerabilities in the TOE using the web sites listed below. The sources of the publicly available information are provided below: • https://nvd.nist.gov/view/vuln.search • http://cve.mitre.org/cve • https://www.cisa.gov • https://docs.progress.com/bundle/release-notes_loadmaster-7-2-54-18/page/Issues-Resolved.html The evaluator performed the public domain vulnerability searches using the following key words. The search was performed on **06/06/2025**, **17/11/2025, 03/23/2026, 04/27/2026 and 17/06/2026**: • Progress Loadmaster • Progress Loadmaster virtual • Progress Loadmaster Physical • LoadMaster • LoadMaster X25-NG • LoadMaster X40-NG • ECS CM H2 NG • ECS CM H3 NG • Intel Xeon Gold 5222 (Cascade Lake) • Intel Xeon Silver 4316 (Ice Lake)

- LoadMaster OS 7.2.54.18
- Libmcrypt v2.5.7
- Syslog ng v3.25.1
- OpenLDAP v2.4.32
- Openssl v1.1.1za
- cpe:/:vmware:esxi:7.0

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

Verdict:
PASS.

7 DETAILED TEST CASES (ACTIVITY)

7.1 TEST ACTIVITIES

7.1.1 SECURITY AUDIT (FAU)

7.1.1.1 FAU_GEN.1 TEST #1 (CPP_ND_V3.0E)

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 identity and authentication (I&A) mechanisms for a system, the FIA_UIA_EXT.1 events must be generated for each mechanism. The evaluator shall test that audit records are generated for the establishment and termination of a channel for each of the cryptographic protocols contained in the ST. If HTTPS is implemented, the test demonstrating the establishment and termination of a TLS session can be combined with the test for an HTTPS session. When verifying the test results, the evaluator shall ensure the audit records generated during testing match the format specified in the guidance documentation, and that the fields in each audit record have the proper entries. Note that the testing here can be accomplished in conjunction with the testing of the security mechanisms directly.
Test Steps	• Trigger each auditable event on the TOE • Verify that each audit record is generated and contains the required information
Pass/Fail with Explanation	Pass. The audit records associated with each test case are recorded. Each required audit record is generated by the TOE. This meets the testing requirement.

7.1.1.2 FAU_GEN.1 TEST #2 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	For distributed TOEs the evaluator shall perform tests on all TOE components according to the mapping of auditable events to TOE components in the Security Target. For all events involving more than one TOE component when an audit event is triggered, the evaluator has to check that the event has been audited on both sides (e.g. failure of building up a secure communication channel between the two components). This is not limited to error cases but includes

	also events about successful actions like successful build up/tear down of a secure communication channel between TOE components. Note that the testing here can be accomplished in conjunction with the testing of the security mechanisms directly.
Pass/Fail with Explanation	N/A. The TOE is not distributed; hence, this activity is not applicable.

7.1.1.3 FAU_GEN.2 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	This activity should be accomplished in conjunction with the testing of FAU_GEN.1.1.
Pass/Fail with Explanation	Pass. This activity was accomplished in conjunction with the testing of FAU_GEN.1.Test #1.

7.1.1.4 FAU_GEN.2 TEST #2 (CPP_ND_V3.0E)

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

7.1.1.5 FAU_STG_EXT.1 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	Testing of secure transmission of the audit data externally (FTP_ITC.1) and, where applicable, intercomponent (FPT_ITT.1 or FTP_ITC.1) shall be performed according to the assurance activities for the particular protocol(s). The evaluator shall perform the following additional test for this requirement:

	Test 1: The evaluator shall establish a session between the TOE and the audit server according to the configuration guidance provided. The evaluator shall then examine the traffic that passes between the audit server and the TOE during several activities of the evaluator's choice designed to generate audit data to be transferred to the audit server. The evaluator shall observe that these data are not able to be viewed in the clear during this transfer, and that they are successfully received by the audit server. The evaluator shall record the particular software (name, version) used on the audit server during testing. The evaluator shall verify that the TOE is capable of transferring audit data to an external audit server automatically without administrator intervention.
Test Steps	• Configure the TOE to send logs to an audit server. • Record the software name and version of the syslog application on the audit server. • Generate the audit event and confirm that each event has been logged on the syslog server. • Verifying the logs seen on the TOE are the same. • Verify that the session was established and encrypted between TOE and audit server via packet capture.
Pass/Fail with Explanation	Pass. TOE is capable of transferring audit data to an external audit server automatically without administrative intervention. This meets the testing requirements.

7.1.1.6 FAU_STG_EXT.1 TEST #2 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	Testing of secure transmission of the audit data externally (FTP_ITC.1) and, where applicable, intercomponent (FPT_ITT.1 or FTP_ITC.1) shall be performed according to the assurance activities for the particular protocol(s). The evaluator shall perform the following additional test for this requirement: Test 2: For distributed TOEs, Test 1 defined above shall be applicable to all TOE components that forward audit data to an external audit server.
Pass/Fail with Explanation	N/A. The TOE is not distributed; hence, this activity is not applicable.

7.1.1.7 FAU_STG_EXT.1 TEST #3 (A) (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	Testing of secure transmission of the audit data externally (FTP_ITC.1) and, where applicable, intercomponent (FPT_ITT.1 or FTP_ITC.1) shall be performed according to the assurance activities for the particular protocol(s). The evaluator shall perform the following additional test for this requirement:

	Test 3: The evaluator shall perform operations that generate audit data and verify that this data is stored locally. The evaluator shall then make note of whether the TSS claims persistent or non-persistent logging and perform one of the following actions: i. If **persistent logging** is selected, the evaluator shall perform a power cycle of the TOE and ensure that following power on operations the log events generated are still maintained within the local audit storage.
Test Steps	• Perform operations on TOE to generate multiple audit logs. • Verify that logs are generated for the above steps. • Verify timestamp of oldest log entry on the TOE. • Reboot the TOE. • Verify oldest log entry is still available on the TOE after reboot.
Pass/Fail with Explanation	Pass. TOE ensures that even after the power cycle the logs are still maintained within the local audit storage of the TOE. This meets the testing requirements.

7.1.1.8 FAU_STG_EXT.1 TEST #3 (B) (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	Testing of secure transmission of the audit data externally (FTP_ITC.1) and, where applicable, intercomponent (FPT_ITT.1 or FTP_ITC.1) shall be performed according to the assurance activities for the particular protocol(s). The evaluator shall perform the following additional test for this requirement: Test 3: The evaluator shall perform operations that generate audit data and verify that this data is stored locally. The evaluator shall then make note of whether the TSS claims persistent or non-persistent logging and perform one of the following actions: ii. If **non-persistent logging** is selected, the evaluator shall perform a power cycle of the TOE and ensure that following power on operations the log events generated are no longer present within the local audit storage.
Pass/Fail with Explanation	N/A, Non-persistent logging selection is not claimed in ST.

7.1.1.9 FAU_STG_EXT.1 TEST #4 (A) (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	Testing of secure transmission of the audit data externally (FTP_ITC.1) and, where applicable, intercomponent (FPT_ITT.1 or FTP_ITC.1) shall be performed according to the assurance activities for the particular protocol(s). The evaluator shall perform the following additional test for this requirement: Test 4: 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.1.5. Depending on the configuration this means that the evaluator shall check the content of the audit data when the audit data is just filled to the maximum and then verifies that: i. 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.1.5).
Test Steps	• Login to the TOE using the xroot access. • Check the disk Usage using df -h. • Now fill up the log partition to about 100% capacity and wait for more than a minute to view the log on the device. • Verify on the TOE that the Disk Usage space is full. • Verify that the disk Usage if full using df -h. • Verify all the log files on TOE stop storing the logs. • Verify that even though the local audit are dropped due to storage full, the new audit logs are still generating and transmitted to the external audit server. • Click on the Clear All option under the Syslog log file option to Free up space from /var/log partition. • Verify that the /var/log file is space has been cleared. • Verify that logging resumed in the TOE.
Pass/Fail with Explanation	Pass. Evaluator verified after the filling the log partition to 100% on the TOE, logs stop getting recorded on the TOE. After freeing local space, logs starts getting recorded on the TOE. Evaluator also verified that while the local storage is full, the TOE still transmits the newly generated logs to the configured external audit server. This meets the test requirements.

7.1.1.10 FAU_STG_EXT.1 TEST #4 (B) (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	Testing of secure transmission of the audit data externally (FTP_ITC.1) and, where applicable, intercomponent (FPT_ITT.1 or FTP_ITC.1) shall be performed according to the assurance activities for the particular protocol(s). The evaluator shall perform the following additional test for this requirement:

	Test 4: 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.1.5. Depending on the configuration this means that the evaluator shall check the content of the audit data when the audit data is just filled to the maximum and then verifies that: ii. 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.1.5)
Pass/Fail with Explanation	N/A. The ST does not select 'overwrite previous audit records'.

7.1.1.11 FAU_STG_EXT.1 TEST #4 (C) (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	Testing of secure transmission of the audit data externally (FTP_ITC.1) and, where applicable, intercomponent (FPT_ITT.1 or FTP_ITC.1) shall be performed according to the assurance activities for the particular protocol(s). The evaluator shall perform the following additional test for this requirement: Test 4: 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.1.5. Depending on the configuration this means that the evaluator shall check the content of the audit data when the audit data is just filled to the maximum and then verifies that: iii. The TOE behaves as specified (for the option '**other action'** in FAU_STG_EXT.1.5).
Pass/Fail with Explanation	N/A, because '**other action'** is not selected in FAU_STG_EXT.1.5.

7.1.1.12 FAU_STG_EXT.1 TEST #5 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	Testing of secure transmission of the audit data externally (FTP_ITC.1) and, where applicable, intercomponent (FPT_ITT.1 or FTP_ITC.1) shall be performed according to the assurance activities for the particular protocol(s). The evaluator shall perform the following additional test for this requirement: Test 5: For distributed TOEs, for the local storage according to FAU_STG_EXT.1.4, Test 1 specified above shall be applied to all TOE components

	that store audit data locally. For all TOE components that store audit data locally and comply with FAU_STG_EXT.2, Test 2 specified above shall be applied. The evaluator shall verify that the transfer of audit data to an external audit server is implemented.
Pass/Fail with Explanation	N/A. The TOE is not distributed; hence, this activity is not applicable.

7.1.1.13 FAU_STG_EXT.1 TEST #6 (CPP_ND_V3.0E) **[TD0886]**

Item	Data
Test Assurance Activity	Testing of secure transmission of the audit data externally (FTP_ITC.1) and, where applicable, intercomponent (FPT_ITT.1 or FTP_ITC.1) shall be performed according to the assurance activities for the particular protocol(s). The evaluator shall perform the following additional test for this requirement: Test 6 [Conditional]: In case manual export or ability to view locally is selected in FAU_STG_EXT.1.6, during interruption the evaluator shall perform a TSF-mediated action and verify the event is recorded in the audit trail. Note: The intent of the test is to ensure that the local audit TSF (as specified by FAU_STG_EXT.1.3) operates independently from the ability to transmit the generated audit data to an external audit server (as specified in FAU_STG_EXT.1.1). There are no specific requirements on the interruption of the connection between the TOE and the external audit server (as for FTP_ITC.1). **TD0886 has been applied.**
Test Steps	• Configure the TOE to send audit logs to an external audit server. • Generate audit events on the TOE. • Verify generated audit logs are stored on the external audit server. • Interrupt the connection between the TOE and external audit server. • Verify that the external audit server does not receive any further logs from the TOE. • Verify that the TOE continues to store logs locally.
Pass/Fail with Explanation	Pass. TOE overwrites the audit records once the local storage space is full. This meets the testing requirements.

7.1.1.14 FAU_STG_EXT.3 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall verify that a warning is issued by the TOE before the local storage space for audit data is full. For distributed TOEs the evaluator shall verify the correct implementation of display warning for local storage space for all TOE components that are supporting this feature according to the description in the TSS. The evaluator shall verify that each component that supports this feature according to the description in the TSS is capable of generating a warning itself or through another component.
Test Steps	• Login to the TOE using the xroot access. • Check the disk usage using df- h. • Update the cron job for log disk usage notification to be executed per minute. • Fill up the log partition to about 90% capacity and wait for more than a minute to view the log on the device. • Verify TOE logs that the local log storage is almost full.
Pass/Fail with Explanation	Pass. The TOE issues a warning before the local log storage is full. This meets the testing requirement.

7.1.2 CRYPTOGRAPHIC SUPPORT (FCS)

7.1.2.1 FCS_CKM.1 RSA (CPP_ND_V3.0E) **[TD0921]**

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 or FIPS PUB 186-5 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. This test must be repeated for each supported RSA modulo and generation method. Key Pair generation specifies 5 ways (or methods) to generate the primes p and q. These include:

a) Random Provable Primes (p and q shall be provable primes)
b) Random Probable primes (p and q shall be probable primes)
c) Provable Primes with Conditions (p1, p2, q1, q2, p and q shall all be provable primes)
d) Provable/probable primes with Conditions (p1, p2, q1, and q2 shall be provable primes and p and q shall be probable primes)
e) Probable primes with Conditions (p1, p2, q1, q2, p and q shall all be probable primes)

The Random Provable primes, and all the Primes with Conditions can be tested in the same manner because each of these begin with a starting random number and calculate the p and q values from this value. The test instructs the TSF to generate intermediate values and the p, q, n, and d values. The evaluator then validates the correctness of the values generated by the TSF.

To test the key generation method for the Random Provable primes method or 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. If the TSF provides the input to the key generation function, such input must be recorded and verified. For each RSA key length (modulo) claimed, the evaluator shall generate 25 key pairs. The evaluator shall verify the correctness of the TSF's implementation by comparing Key Pair values generated by the TSF with those generated using the same set of input values using a known good implementation.

The Random Probable primes must be tested in a different way because this test generates different random numbers, not related to each other, until the number satisfies the "probably prime" requirements. This validation method requires two tests for Random Probable primes. These include the Known Answer Test and the Miller-Rabin probabilistic primality test.

To test the key generation method for the Random Probable primes, the known answer test must be used. The evaluator shall compare the results of a known good implementation with the TSF results for a set (of a corresponding size depending on modulus) of supplied values containing private prime factor p, private prime factor q and confirm all public keys, e, match. Then the evaluator shall use the TSF to generate prime p, q pairs for each modulus size and perform the Miller-Rabin tests.

TD0921 has been applied.

Pass/Fail with Explanation	N/A. RSA Keygen is not claimed in the ST.

7.1.2.2 FCS_CKM.1 ECC (CPP_ND_V3.0E) **[TD0921]**

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)** *Key Generation for ECDSA Schemes* Key pairs for the ECDSA consist of pairs (d, Q), where the private key, d, is an integer, and the public key, Q, is an elliptic curve point. The evaluator shall verify the implementation of ECC Key Generation by the TOE using the following components tests: • ECC Key Pair Generation • ECC Public Key Validation (PKV) *ECC Key Pair Generation Test* There are four methods by which these pairs may be generated: • FIPS 186-4 Section B.4.1 Key Pair Generation Using Extra Random Bits Using this method, 64 more bits are requested from the RBG than are needed for d so that bias produced by the mod function is negligible. • FIPS 186-4 Section B.4.2 Key Pair Generation by Testing Candidates Using this method, a random number is obtained and tested to determine that it will produce a value of d in the correct range. If d is out-of-range, another random number is obtained (i.e., the process is iterated until an acceptable value of d is obtained. • FIPS 186-5 Section A.2.1 ECDSA Key Pair Generation using Extra Random Bits Using this method, more bits are requested from the DRBG than are needed for d so that the bias produced by the mod function is negligible. • FIPS 186-5 Section A2.2 ECDSA Key Pair Generation by Rejection Sampling Using this method, a random number is obtained and tested to determine that it will produce a value of d in the correct range. If d is out of range, an ERROR is returned. The evaluator shall test the ECC Key Pair Generation by having the TSF produce 10 key pairs (d, Q) for each implemented key generation method using each supported curve (i.e., P-256, P-384, and P-521). The steps are the same for each key generation method

	and curve. The private key, d, shall be generated using the output of an approved DRBG converted to an integer via modular reduction or the discard method. The known private key is then used by the TSF to compute the public key, Q'. To evaluator then validates the correctness by comparing the value Q' computed by the TSF to the public key, Q generated by a known good implementation. *ECC Public Key Verification (PKV) Test* The evaluator shall generate 12 key pairs (d, Q) for each selected curve, with 6 valid public keys, Q, using a known good implementation and 6 modified, Q', and determine whether the TSF can accurately detect these modifications. Q' should be otherwise valid but include at least one of the following errors: a) point X or Y not on the curve, b) point X or Y is too large for the field for the given curve. The evaluator encouraged to make sure that the modification does not inadvertently result in another valid public key (e.g., modifying Y and accidentally hitting a different point on the curve). **TD0921 has been applied.**
Pass/Fail with Explanation	Algorithm: ECDSA KeyGen (FIPS186-5) Curves: P-256, P-384, P-521 CAVP #: A7300 Algorithm: ECDSA KeyVer (FIPS186-5) Curves: P-256, P-384, P-521 CAVP #: A7300 Pass. Based on these findings, this assurance activity is considered satisfied.

7.1.2.3 FCS_CKM.1 FFC - FIPS PUB 186-4 (CPP_ND_V3.0E)

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.
Pass/Fail with Explanation	N/A. FFC is not claimed in the ST.

7.1.2.4 FCS_CKM.1 FFC - "SAFE-PRIME" GROUPS (CPP_ND_V3.0E)

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). **FFC Schemes using "safe-prime" groups** Testing for FFC Schemes using safe-prime groups is done as part of testing in CKM.2.1.
Pass/Fail with Explanation	N/A. FFC is not claimed in the ST.

7.1.2.5 FCS_CKM.2 RSA (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	**RSA-based key establishment** The evaluator shall verify the correctness of the TSF's implementation of RSAES-PKCS1-v1_5 by using a known good implementation for each protocol selected in FTP_TRP.1/Admin, FTP_TRP.1/Join, FTP_ITC.1 and FPT_ITT.1 that uses RSAES-PKCS1-v1_5.
Pass/Fail with Explanation	N/A. RSA is not claimed in the ST.

7.1.2.6 FCS_CKM.2 SP800-56A - ECC (CPP_ND_V3.0E)

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. **ECC and FIPS 186-type FFC 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 for ECC and FIPS186-type. 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 shall 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 CAVP #: A7300 Pass. Based on these findings, this assurance activity is considered satisfied.

7.1.2.7 FCS_CKM.2 SP800-56A - FFC (CPP_ND_V3.0E)

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. **ECC and FIPS 186-type FFC 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 for ECC and FIPS186-type. 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 shall 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	N/A. KeyGen for FFC is not claimed as per ST.

7.1.2.8 FCS_CKM.2 FFC SAFE-PRIME (CPP_ND_V3.0E)

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	N/A. FFC safe-prime is not claimed in the ST.

7.1.2.9 FCS_CKM.4 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	There are no test assurance activities.
Pass/Fail with Explanation	N/A. There are no test assurance activities for this SFR. Based on these findings, this assurance activity is considered satisfied.

7.1.2.10 FCS_COP.1/DATAENCRYPTION AES-CBC (CPP_ND_V3.0E)

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 AES-CBC 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 #: A7300 Pass. Based on these findings, this assurance activity is considered satisfied.

7.1.2.11 FCS_COP.1/DATAENCRYPTION AES-GCM (CPP_ND_V3.0E)

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 non-zero 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 #: A7300 Pass. Based on these findings, this assurance activity is considered satisfied.

7.1.2.12 FCS_COP.1/DATAENCRYPTION AES-CTR (CPP_ND_V3.0E)

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 if the TSF is validated against the requirements of the Functional Package for Secure Shell referenced in Section 2.2 of the cPP. 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, AES-GCM 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: 256 CAVP #: A7300 Pass. Based on these findings, this assurance activity is considered satisfied.

7.1.2.13 FCS_COP.1/SIGGEN ECDSA (CPP_ND_V3.0E) **[TD0921]**

Item	Data
Test Assurance Activity	**ECDSA Signature Algorithm Tests** The evaluator shall verify the implementation of ECDSA Signature generation by the TOE using the Signature Generation Test. This test validates the TSF generation of the (r, s) pair that represent the digital signature. The digital signature shall be verified using the same domain parameters and hash function that were used during signature generation. An approved hash function or an XOF shall be used for this purpose. **Signature Generation Test** The purpose of this test is to verify the ability of the TSF to produce correct signatures. To test signature generation, the evaluator supplies 10 pseudorandom messages to the TSF and a key pair, (d, Q), generated by a known good implementation. Exercising each applicable curve (i.e., P-256, P 384, or P-521) and hash algorithm or extendable-output function combination, the TSF generates signatures for each supplied message and returns the corresponding signatures. Using a known-good implementation, the evaluator validates the signatures by using the associated public key, Q, to verify the signature. **Signature Verification Test**

	The purpose of this test is to verify the ability of the TSF to accept valid signatures and reject invalid signatures. For each curve/hash algorithm or extendable-output function combination supported by the TSF, the evaluator shall use a known good implementation to generate a key pair, (d, Q), and use known good implementation with the private key, d, to sign 15 pseudorandom messages of 1024 bits. The evaluator shall alter some of the messages or signatures so that signature verification should fail. To test signature verification, the evaluator supplies the public key, Q, and 15 pseudorandom messages, including altered messaged, to the TSF for verification of signatures. The evaluator shall then verify that the TSF validates correct signatures on the original messages and flags or rejects the altered messages. **TD0921 has been applied.**
Pass/Fail with Explanation	Algorithm: ECDSA SigGen (FIPS186-5) Curves: P-256, P-384, P-521 CAVP #: A7300 Algorithm: ECDSA SigVer (FIPS186-5) Curves: P-256, P-384, P-521 CAVP #: A7300 Pass. Based on these findings, this assurance activity is considered satisfied.

7.1.2.14 FCS_COP.1/SIGGEN RSA (CPP_ND_V3.0E) **[TD0921]**

Item	Data
Test Assurance Activity	**RSA Signature Algorithm Tests** **Signature Generation Test** The evaluator shall verify the implementation of RSA Signature generation by the TOE using the Signature Generation Test. This test verifies the ability of the TSF to produce correct signatures. There are 2 different RSA Signature algorithms that can be implemented. These include: a. RSASSA-PKCS1-v1.5 b. RSASSA-PSS

	To test signature generation, the evaluator generates or obtains 10 messages for each modulus size/hash or extendable-output function combination supported by the TOE. Using a key generated by a known good implementation, the TSF generates and returns the corresponding signatures to a known good implementation that validates the signatures by using the associated public key to verify the signature. 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** The evaluator shall verify the implementation of RSA Signature verification by the TOE using the Signature Verification Test. This test verifies the ability of the TSF to recognize valid and invalid signatures. There are 2 different RSA Signature algorithms that can be implemented. These include: a. RSASSA-PKCS1-v1.5 b. RSASSA-PSS For each modulus size/hash or extendable-output function combination supported by the TOE, the evaluator shall use a known good implementation to generate a modulus and three associated key pairs, (d, e). Each private key d is used to sign six pseudorandom messages each of 1024 bits. Some of the public keys, e, messages, IR format, or signatures must be altered so that signature verification should fail. The modifications must cover distinct "key modified", "message modified", "signature modified", "IR moved", and "trailer moved" tests. The modulus, hash or extendable-output algorithm, public key e values, messages, and signatures are forwarded to the TSF. The TSF then attempts to verify the signatures and returns the results. The evaluator then compares the received results with the stored results from a known good implementation. The evaluator verifies that the TSF validates correct signatures on the original messages and flags or rejects the altered messages. **TD0921 has been applied.**
Pass/Fail with Explanation	Algorithm: RSA SigGen (FIPS186-5) Key size / Modulus: 2048 CAVP #: A7300

	Algorithm: RSA SigVer (FIPS186-5) Key size / Modulus: 2048 CAVP #: A7300 Pass. Based on these findings, this assurance activity is considered satisfied.

7.1.2.15 FCS_COP.1/HASH (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The TSF hashing functions can be implemented in one of two modes. The first mode is the byteoriented 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 bitoriented 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 bitoriented vs. the byteoriented 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 evaluator shall 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 evaluator shall 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 evaluator shall 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 evaluator shall 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 byteoriented implementations only. The evaluator shall 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 evaluator shall 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 #: A7300 Pass. Based on these findings, this assurance activity is considered satisfied.

7.1.2.16 FCS_COP.1/KEYEDHASH (CPP_ND_V3.0E)

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) CAVP #: A7300 Pass. Based on these findings, this assurance activity is considered satisfied.

7.1.2.17 FCS_RBG_EXT.1 (CPP_ND_V3.0E)

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 shall verify 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 shall verify 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 (CTR_DRBG) Mode: AES-256 CAVP #: A7300 Pass. Based on these findings, this assurance activity is considered satisfied.

7.1.2.18 FCS_HTTPS_EXT.1 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	This test is now performed as part of FIA_X509_EXT.1/Rev testing. Tests are performed in conjunction with the TLS evaluation activities. If the TOE is an HTTPS client or an HTTPS server utilizing X.509 client authentication, then the certificate validity shall be tested in accordance with testing performed for FIA_X509_EXT.1.
Pass/Fail with Explanation	N/A. TOE is a HTTPS server without mutual authentication.

7.1.2.19 FCS_NTP_EXT.1.1 TEST #1 (CPP_ND_V3.0E)

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	• Configure the NTP server with supported message digest algorithm. • Connect the TOE to an NTP server and verify that the synchronization succeeds. • Verify the connection was established via packet capture. • Verify the connection was established via logs.
Pass/Fail with Explanation	Pass. The TOE uses the correct NTP version specified in the ST. This meets the testing requirement.

7.1.2.20 FCS_NTP_EXT.1.2 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The cryptographic algorithms selected in element 1.2 and specified in the ST will have been specified in an FCS_COP SFR and tested in the accompanying Evaluation Activity for that SFR. Likewise, the cryptographic protocol selected in in element 1.2 and specified in the ST will have been specified in an FCS SFR and tested in the accompanying Evaluation Activity for that SFR. [conditional] If the **message digest algorithm** is claimed in element 1.2, the evaluator shall 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 shall use the captured network traffic, to verify the NTP version, to observe time change of the TOE and uses the TOE's audit log to determine that the TOE accepted the NTP server's timestamp update. The captured traffic is also used to verify that the appropriate message digest algorithm was used to authenticate the time source and/or the appropriate protocol was used to ensure integrity of the timestamp that was transmitted in the NTP packets.
Test Steps	NTP connection with supported message digest algorithm The testing for this is covered in FCS_NTP_EXT.1.1 TEST #1. NTP connection with unsupported message digest algorithm • Configure the NTP server with unsupported message digest algorithm. • Connect the TOE to an NTP server and verify that the synchronization fails. • Verify via logs that the connection was refused. • Verify the connection was refused via packet capture.
Pass/Fail with Explanation	Pass. The TOE is in sync with NTP Server when supported Message Digest algorithm is configured on NTP Server, this satisfies the requirement.

7.1.2.21 FCS_NTP_EXT.1.3 TEST #1 (CPP_ND_V3.0E)

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: • Configure an NTP server to update from a broadcast address. • Configure TOE to sync with the NTP server. • Verify that the TOE does not accept broadcast updates from the NTP server. • Verify with packet capture. Multicast: • Configure an NTP server to update from a multicast address. • Configure TOE to sync with the NTP server.

	• Verify with packet capture that the TOE does not accept multicast updates from the NTP server. • Check new time. There have been no significant changes other than normal passage of time.
Pass/Fail with Explanation	Pass. The TOE does not sync with an NTP server that sends out broadcast and multicast updates. This meets testing requirements.

7.1.2.22 FCS_NTP_EXT.1.4 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 1: The evaluator shall confirm the TOE supports configuration of at least three (3) NTP time sources. The evaluator shall configure at least three NTP servers to support periodic time updates to the TOE. The evaluator shall confirm the TOE is configured to accept NTP packets that would result in the timestamp being updated from each of the NTP servers. The evaluator shall check that the time stamp is updated after receipt of the NTP packets. The purpose of this test to verify that the TOE can be configured to synchronize with multiple NTP servers. It is up to the evaluator to determine that the multi- source update of the time information is appropriate and consistent with the behaviour prescribed by the RFC 1305 for NTPv3 and RFC 5905 for NTPv4.
Test Steps	• Configure at least 3 NTP time sources on TOE. • Configure the NTP server to use the same key ID. • Start the NTP synchronization and verify that it syncs to NTP server with IP 10.1.4.76. • Verify via packet capture that the TOE has synced to NTP server with IP 10.1.4.76. • Verify via logs that the TOE has synced to NTP server with IP 10.1.4.76. • Stop the previous NTP server and sync with another NTP server with IP 10.1.4.42. • Configure this NTP server to use the same key ID. • Start the NTP server. • Start the NTP synchronization and verify that it syncs to NTP server with IP 10.1.4.42. • Verify via packet capture that the TOE has synced to NTP server with IP 10.1.4.42. • Verify via logs that the TOE has synced to NTP server with IP 10.1.4.42. • Stop the previous NTP server to sync with another NTP server with IP 10.1.4.58. • Configure this NTP server to use the same key ID. • Start the NTP server. • Start the NTP synchronization and verify that it syncs to NTP server with IP 10.1.4.58.

	• Verify via packet capture that the TOE has synced with NTP server with IP 10.1.4.58. • Verify via logs that the TOE has synced with NTP server with IP 10.1.4.58. • Verify that that the multi- source update of the time information is appropriate and consistent with the behavior prescribed by the RFC 5905
Pass/Fail with Explanation	Pass. The TOE successfully handles the use of multiple NTP servers. This meets testing requirements.

7.1.2.23 FCS_NTP_EXT.1.4 TEST #2 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: 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 response 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. It is up to the evaluator to craft and transmit unsolicited updates in a way that would be consistent with the behaviour of a correctly-functioning NTP server.
Test Steps	• Verify the time on the TOE. • Configure an NTP server. • Sync the TOE with NTP server and capture those packets. • Verify with packet capture. • Configure a different NTP server to which the TOE syncs. • Replay the packets from the NTP server which were captured during earlier sync. • Verify the TOE does not sync with the NTP server.
Pass/Fail with Explanation	Pass. The TOE only accepts NTP updates from configured NTP Servers and rejects NTP updates from unconfigured NTP Servers. This meets the testing requirements.

7.1.2.24 FCS_TLSC_EXT.1.1 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests:

	Test 1: The evaluator shall establish a TLS connection using each of the ciphersuites specified by the requirement. This connection may be established as part of the establishment of a higher-level protocol, e.g., as part of an HTTPS session. It is sufficient to observe the successful negotiation of a ciphersuite to satisfy the intent of the test; it is not necessary to examine the characteristics of the encrypted traffic to discern the ciphersuite being used (for example, that the cryptographic algorithm is 128-bit AES and not 256-bit AES).
Pass/Fail with Explanation	Testing for each claimed cipher was performed in FCS_TLSC_EXT.1.6 Test #1. This meets the testing requirements.

7.1.2.25 FCS_TLSC_EXT.1.1 TEST #2 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: The goal of the following test is to verify that the TOE accepts only certificates with appropriate values in the extendedKeyUsage extension, and implicitly that the TOE correctly parses the extendedKeyUsage extension as part of X.509v3 server certificate validation. Test 2: The evaluator shall establish the connection with a server presenting a certificate that contains the serverAuth (OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage extension and verify that the connection successfully negotiated. The evaluator shall then verify that when the same server presents an otherwise valid server certificate that contains the extendedKeyUsage extension without serverAuth the client rejects the connection. Ideally, the two certificates should be identical except for the OID values.
Test Steps	<u>Valid Certificate</u> • Create a server certificate that contains the Server Authentication purpose in the extendedKeyUsage extension. • Utilize acumen-tls tool to start a TLS server and ensure the connection was accepted. • Verify the successful TLS connection via packet capture. <u>Invalid Certificate</u> • Create a server certificate missing the Server Authentication purpose in the extendedKeyUsage extension. • Utilize acumen-tls tool to start a TLS server and ensure the connection was rejected.

	• Verify the unsuccessful TLS connection via TOE logs. • Verify the unsuccessful TLS connection via packet capture.
Pass/Fail with Explanation	Pass. TOE rejects server certificates lacking the Server Authentication purpose in the extended key usage extension and accepts server certificates having Server Authentication purpose in the extended key usage extension. This meets the testing requirements.

7.1.2.26 FCS_TLSC_EXT.1.1 TEST #3 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 3 [conditional]: Perform this test only if support of TLS 1.2 is claimed. The evaluator shall send a server certificate in the TLS connection that does not match the server-selected ciphersuite (for example, send an 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	• Utilize Acumen-tls tool to send a server certificate that does not match the server-selected ciphersuite and ensure the connection was rejected. • Verify the unsuccessful TLS connection via TOE logs. • Verify the unsuccessful TLS connection due to ciphersuite mismatch using packet capture.
Pass/Fail with Explanation	Pass. TOE rejects server certificates not matching server-selected ciphersuite. This meets the testing requirements.

7.1.2.27 FCS_TLSC_EXT.1.1 TEST #4A (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 4: The evaluator shall perform the following 'negative tests': i. [conditional]: Perform this test only if support of TLS 1.2 is claimed. The evaluator shall configure the server to select the TLS_NULL_WITH_NULL_NULL ciphersuite and verify that the TOE TLS client denies the connection.
Test Steps	• Utilize Acumen-tls tool to start a server with TLS_NULL_WITH_NULL_NULL ciphersuite and ensure the connection was rejected. • Verify the unsuccessful TLS connection via TOE logs.

	• Verify the unsuccessful TLS connection via packet capture.
Pass/Fail with Explanation	Pass. TOE rejects server certificates using TLS_NULL_WITH_NULL_NULL ciphersuite. This meets the testing requirements.

7.1.2.28 FCS_TLSC_EXT.1.1 TEST #4B (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 4: The evaluator shall perform the following 'negative tests': ii. Modify the server's selected ciphersuite in the Server Hello handshake message to be a ciphersuite (compatible with the server-selected version of TLS) not presented in the Client Hello handshake message. The evaluator shall verify that the TOE TLS client rejects the connection after receiving the Server Hello.
Test Steps	• Utilize Acumen-tls tool to start a server with an unsupported ciphersuite and ensure the connection was rejected. • Verify the unsuccessful TLS connection via TOE logs. • Verify the unsuccessful TLS connection via packet capture.
Pass/Fail with Explanation	Pass. TOE rejects server certificates using an unsupported ciphersuite. This meets the testing requirements.

7.1.2.29 FCS_TLSC_EXT.1.1 TEST #4C (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 4: The evaluator shall perform the following 'negative tests': The evaluator shall attempt to establish a TLS connection using each valid TLS/SSL version (i.e. TLS 1.3, TLS 1.2, TLS 1.1, TLS 1.0, SSL 3.0, SSL 2.0). The evaluator shall verify that the version(s) specified in FCS_TLSC_EXT.1.1 are successfully established and all other versions are rejected by the TOE TLS client. If a supported_versions extension is not sent by the TOE in the ClientHello, then the evaluator must ensure the test server responds with a ServerHello that is valid for the TLS version being negotiated. If the TOE includes the Supported Versions extension in its ClientHello, the evaluator shall also ensure the version(s) specified in the extension match the version(s) in FCS_TLSC_EXT.1.1. NOTE: For TLS 1.3 aware test servers, it is appropriate for the test server to issue a TLS Alert. The TOE client must not attempt to continue the connection.

Test Steps	• Utilize Acumen-tls tool to start a TLS server using TLS version 1.2 and verify the connection was successful. • Verify that the TLS session was established for TLS version 1.2 via packet capture. • Utilize Acumen-tls tool to start a TLS server using TLS version 1.3 and verify the connection failure. • Verify that the TLS session was not established for TLS version 1.3 via log • Verify that the TLS session was not established for TLS version 1.3 via packet capture. • Utilize Acumen-tls tool to start a TLS server using TLS version 1.1 and verify the connection failure. • Verify that the TLS session was not established for TLS version 1.1 via log. • Verify that the TLS session was not established for TLS version 1.1 via packet capture. • Utilize Acumen-tls tool to start a TLS server using TLS version 1.0 and verify the connection failure. • Verify that the TLS session was not established for TLS version 1.0 via log • Verify that the TLS session was not established for TLS version 1.0 via packet capture. • Utilize Acumen-tls tool to start a TLS server using SSL version 3.0 and verify the connection failure. • Verify that the TLS session was not established for SSL version 3.0 via log. • Verify that the TLS session was not established for SSL version 3.0 via packet capture. • Utilize Acumen-tls tool to start a TLS server using SSL version 2.0 and verify the connection failure. • Verify that the TLS session was not established for SSL version 2.0 via log. • Verify that the TLS session was not established for SSL version 2.0 via packet capture.
Pass/Fail with Explanation	Pass. TOE rejects the TLS connection for each unclaimed TLS/SSL version and accepts the TLS connection for each claimed TLS version. This meets the testing requirements.

7.1.2.30 FCS_TLSC_EXT.1.1 TEST #5A (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 5: The evaluator shall perform the following modifications to the traffic (i.e. Man-in-the-middle modifications that result in invalid signatures and MACs):

	[conditional]: Perform this test only if support of TLS 1.2 is claimed. If using DHE or ECDH ciphersuites, modify the signature block in the Server's Key Exchange handshake message, and verify that the client denies the connection and no application data flows. The handshake shall be valid (e.g. the Finished message is calculated using the modified signature), with the exception of the invalid signature. This test does not apply to ciphersuites using RSA key exchange. If a TOE only supports RSA key exchange in conjunction with TLS, then this test shall be omitted.
Test Steps	• Utilize Acumen-tls tool to modify the signature in the Server Key Exchange and verify the connection failure. • Verify the unsuccessful TLS connection via TOE logs. • Verify the unsuccessful TLS connection via packet capture.
Pass/Fail with Explanation	Pass. TOE rejects the TLS connection for the server certificate due to a modified signature block in the Server's Key Exchange handshake message. This meets the testing requirements.

7.1.2.31 FCS_TLSC_EXT.1.1 TEST #5B (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 5: The evaluator shall perform the following modifications to the traffic (i.e. Man-in-the-middle modifications that result in invalid signatures and MACs): [conditional]: Perform this test only if support of TLS 1.3 is claimed. Modify the signature block in the Server's Certificate Verify handshake message, and verify that the client denies the connection and no application data flows. The handshake shall be valid (e.g. the Finished message is calculated using the modified signature), with the exception of the invalid signature.
Pass/Fail with Explanation	N/A. Support for TLS version 1.3 is not claimed by ST. This meets the testing requirements.

7.1.2.32 FCS_TLSC_EXT.1.1 TEST #6A (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 6: The evaluator shall perform the following 'scrambled message tests': i. Modify a byte in the Server Finished handshake message and verify that the handshake does not finish successfully and no application data flows.

	(Note: This modification must be performed prior to the contents of the Finished message being encrypted.)
Test Steps	• Utilize Acumen-tls tool to start a server that will modify a byte in the Server Finished message and verify the connection was unsuccessful. • Verify the unsuccessful TLS connection via TOE logs. • Verify the unsuccessful TLS connection via packet capture.
Pass/Fail with Explanation	Pass. TOE rejects the TLS connection for the server certificate due to the modification in the Server Finished message. This meets the testing requirements.

7.1.2.33 FCS_TLSC_EXT.1.1 TEST #6B (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 6: The evaluator shall perform the following 'scrambled message tests': [conditional]: Perform this test only if support of TLS 1.2 is claimed. Send a garbled message from the server after the server has issued the ChangeCipherSpec message and verify that the handshake is not finished successfully and no application data flows. (Note: TLS 1.3 provides for a dummy ChangeCipherSpec message to aid in middlebox compatibility if such an option is enabled in the specific implementation [see Section D.4 in RFC 8446]. If TLS 1.3 middlebox compatibility mode is enabled a ChangeCipherSpec message may appear in packet traces, but it does not influence the protocol. To be clear: for TLS 1.3, this test does not need to be performed.)
Test Steps	• Utilize Acumen-tls tool to start a server that will send a garbled message after the ChangeCipherSpec message and verify the connection was unsuccessful. • Verify the unsuccessful TLS connection via TOE logs. • Verify the unsuccessful TLS connection via packet capture.
Pass/Fail with Explanation	Pass. TOE rejects the TLS connection with the server sending a garbled message after the server has issued the ChangeCipherSpec message. This meets the testing requirements.

7.1.2.34 FCS_TLSC_EXT.1.1 TEST #6C (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 6: The evaluator shall perform the following 'scrambled message tests':

	ii. [conditional]: Perform this test only if support of TLS 1.3 is claimed. Send a plaintext EncryptedExtensions message from the server and verify that the handshake is not finished successfully and no application data flows. (Note: Under TLS 1.3, the EncryptedExtensions message is the first message to be encrypted with the handshake traffic secret.)
Pass/Fail with Explanation	N/A. Support for TLS version 1.3 is not claimed by ST.

7.1.2.35 FCS_TLSC_EXT.1.1 TEST #6D (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 6: The evaluator shall perform the following 'scrambled message tests': iii. [conditional]: Perform this test only if support of TLS 1.2 is claimed. 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	• Utilize Acumen-tls tool to modify a byte in the server nonce and verify the connection was unsuccessful. • Verify the unsuccessful TLS connection via TOE logs. • Verify the unsuccessful TLS connection via packet capture.
Pass/Fail with Explanation	Pass. TOE rejects the TLS connection with the server due to the modified server's nonce in the Server Hello handshake message. This meets the testing requirements.

7.1.2.36 FCS_TLSC_EXT.1.2 TEST #1 (CPP_ND_V3.0E)

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 1 [conditional]: 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.
Note	*Note that the following tests are marked conditional and are applicable under the following conditions:* *a) For TLS-based trusted channel communications according to FTP_ITC.1 where RFC 6125 is selected, tests 1-6 are applicable.* *or* *b) For TLS-based trusted path communications according to FTP_TRP where RFC 6125 is selected, tests 1-6 are applicable.* *or* *c) For TLS-based trusted path communications according to FPT_ITT.1 where RFC 6125 is selected, tests 1-6 are applicable. Where RFC 5280 is selected, only test 7 is applicable.* *Note that for some tests additional conditions apply.* *IP addresses are binary values that must be converted to a textual representation when presented in the CN of a certificate. When testing IP addresses in the CN, the evaluator shall follow the following formatting rules:* • *IPv4: The CN contains a single address that is represented a 32-bit numeric address (IPv4) is written in decimal as four numbers that range from 0-255 separated by periods as specified in RFC 3986.* • *IPv6: The CN contains a single IPv6 address that is represented as eight colon separated groups of four lowercase hexadecimal digits, each group representing 16 bits as specified in RFC 4291. Note: Shortened addresses, suppressed zeros, and embedded IPv4 addresses are not tested.*
Test Steps	<u>IPv4</u> • Configure the hostname on the TOE. • Create a certificate with incorrect CN and missing SAN. • Utilize acumen-tls tool to start a TLS server with certificate which has incorrect CN and missing SAN and verify the connection was rejected. • Verify via TOE logs that connection was rejected. • Verify via packet capture that connection was rejected. <u>FQDN</u> • Configure hostname on the TOE. • Create a certificate with incorrect CN and missing SAN. • Utilize acumen-tls tool to start a TLS server with certificate which has incorrect CN and missing SAN and verify the connection was rejected. • Verify via TOE logs that connection was rejected. Verify via packet capture that connection was rejected.

Pass/Fail with Explanation	Pass. The TOE rejects the TLS connection for the server certificate that contains a CN that does not match the reference identifier and does not contain the SAN extension. This meets the testing requirements.

7.1.2.37 FCS_TLSC_EXT.1.2 TEST #2 (CPP_ND_V3.0E)

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 2 [conditional]: 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.
Note	*Note that the following tests are marked conditional and are applicable under the following conditions:* *a) For TLS-based trusted channel communications according to FTP_ITC.1 where RFC 6125 is selected, tests 1-6 are applicable.* *or* *b) For TLS-based trusted path communications according to FTP_TRP where RFC 6125 is selected, tests 1-6 are applicable.* *or* *c) For TLS-based trusted path communications according to FPT_ITT.1 where RFC 6125 is selected, tests 1-6 are applicable. Where RFC 5280 is selected, only test 7 is applicable.* *Note that for some tests additional conditions apply.* *IP addresses are binary values that must be converted to a textual representation when presented in the CN of a certificate. When testing IP addresses in the CN, the evaluator shall follow the following formatting rules:* • *IPv4: The CN contains a single address that is represented a 32-bit numeric address (IPv4) is written in decimal as four numbers that range from 0-255 separated by periods as specified in RFC 3986.* • *IPv6: The CN contains a single IPv6 address that is represented as eight colon separated groups of four lowercase hexadecimal digits, each group representing 16 bits as specified in RFC 4291. Note: Shortened addresses, suppressed zeros, and embedded IPv4 addresses are not tested.*
Test Steps	

	SAN: IPv4 • Configure the hostname on the TOE. • Create a certificate with the correct CN and incorrect SAN. • Utilize acumen-tls tool to start a TLS server with certificate which has correct CN and incorrect SAN and verify the connection was rejected. • Verify via TOE logs that connection was unsuccessful. • Verify via packet capture that connection was unsuccessful. SAN: FQDN • Configure the hostname on the TOE. • Create a certificate with the correct CN and incorrect SAN. • Utilize acumen-tls tool to start a TLS server with certificate which has correct CN and incorrect SAN and verify the connection was rejected. • Verify via TOE logs that connection was unsuccessful. Verify via packet capture that connection was unsuccessful.
Pass/Fail with Explanation	Pass. TOE rejects a server certificate that contains the SAN extension but does not contain an identifier in the SAN that matches the reference identifier configured on the TOE for each claimed SAN type (FQDN, IPv4, and IPv6). This meets the testing requirements.

7.1.2.38 FCS_TLSC_EXT.1.2 TEST #3 (CPP_ND_V3.0E)

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 3 [conditional]: 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.
Note	*Note that the following tests are marked conditional and are applicable under the following conditions:* *a) For TLS-based trusted channel communications according to FTP_ITC.1 where RFC 6125 is selected, tests 1-6 are applicable.* *or* *b) For TLS-based trusted path communications according to FTP_TRP where RFC 6125 is selected, tests 1-6 are applicable.* *or*

	c) *For TLS-based trusted path communications according to FPT_ITT.1 where RFC 6125 is selected, tests 1-6 are applicable. Where RFC 5280 is selected, only test 7 is applicable.* *Note that for some tests additional conditions apply.* *IP addresses are binary values that must be converted to a textual representation when presented in the CN of a certificate. When testing IP addresses in the CN, the evaluator shall follow the following formatting rules:* • *IPv4: The CN contains a single address that is represented a 32-bit numeric address (IPv4) is written in decimal as four numbers that range from 0-255 separated by periods as specified in RFC 3986.* • *IPv6: The CN contains a single IPv6 address that is represented as eight colon separated groups of four lowercase hexadecimal digits, each group representing 16 bits as specified in RFC 4291. Note: Shortened addresses, suppressed zeros, and embedded IPv4 addresses are not tested.*
Test Steps	IPv4 • Configure the hostname on the TOE. • Create a certificate with correct CN and missing SAN. • Utilize acumen-tls tool to start a TLS server with certificate which has correct CN and missing SAN and verify the connection was successful. • Verify via packet capture that connection was successful. FQDN: • Configure the hostname on the TOE. • Create a certificate with correct CN and missing SAN. • Utilize acumen-tls tool to start a TLS server with certificate which has correct CN and missing SAN and verify the connection was successful. • Verify via packet capture that connection was successful.
Pass/Fail with Explanation	Pass. The TOE accepts the TLS connection for the server certificate that contains a CN that matches the reference identifier configured on the TOE and does not contain the SAN extension. This meets the testing requirements.

7.1.2.39 FCS_TLSC_EXT.1.2 TEST #4 (CPP_ND_V3.0E)

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 4 [conditional]: 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).

Note	*Note that the following tests are marked conditional and are applicable under the following conditions:* *a) For TLS-based trusted channel communications according to FTP_ITC.1 where RFC 6125 is selected, tests 1-6 are applicable.* *or* *b) For TLS-based trusted path communications according to FTP_TRP where RFC 6125 is selected, tests 1-6 are applicable.* *or* *c) For TLS-based trusted path communications according to FPT_ITT.1 where RFC 6125 is selected, tests 1-6 are applicable. Where RFC 5280 is selected, only test 7 is applicable.* *Note that for some tests additional conditions apply.* *IP addresses are binary values that must be converted to a textual representation when presented in the CN of a certificate. When testing IP addresses in the CN, the evaluator shall follow the following formatting rules:* • *IPv4: The CN contains a single address that is represented a 32-bit numeric address (IPv4) is written in decimal as four numbers that range from 0-255 separated by periods as specified in RFC 3986.* • *IPv6: The CN contains a single IPv6 address that is represented as eight colon separated groups of four lowercase hexadecimal digits, each group representing 16 bits as specified in RFC 4291. Note: Shortened addresses, suppressed zeros, and embedded IPv4 addresses are not tested.*
Test Steps	<u>SAN: IPv4</u> • Configure the hostname on the TOE. • Create a certificate with the incorrect CN and correct SAN. • Utilize acumen-tls tool to start a TLS server with certificate which has incorrect CN and correct SAN and verify the connection was successful. • Verify via packet capture that the connection was successful. <u>SAN: FQDN</u> • Configure hostname on the TOE. • Create a certificate with the incorrect CN and correct SAN. • Utilize acumen-tls tool to start a TLS server with certificate which has incorrect CN and correct SAN and verify the connection was successful. • Verify via packet capture that the connection was successful.
Pass/Fail with Explanation	Pass. TOE accept a TLS connection for 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 reference identifier configured on the TOE. This meets the testing requirements.

7.1.2.40 FCS_TLSC_EXT.1.2 TEST #5 (1) (CPP_ND_V3.0E)

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 5 [conditional]: 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): i. [conditional]: 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.
Note	*Note that the following tests are marked conditional and are applicable under the following conditions:* *a) For TLS-based trusted channel communications according to FTP_ITC.1 where RFC 6125 is selected, tests 1-6 are applicable.* *or* *b) For TLS-based trusted path communications according to FTP_TRP where RFC 6125 is selected, tests 1-6 are applicable.* *or* *c) For TLS-based trusted path communications according to FPT_ITT.1 where RFC 6125 is selected, tests 1-6 are applicable. Where RFC 5280 is selected, only test 7 is applicable.* *Note that for some tests additional conditions apply.* *IP addresses are binary values that must be converted to a textual representation when presented in the CN of a certificate. When testing IP addresses in the CN, the evaluator shall follow the following formatting rules:* • *IPv4: The CN contains a single address that is represented a 32-bit numeric address (IPv4) is written in decimal as four numbers that range from 0-255 separated by periods as specified in RFC 3986.* • *IPv6: The CN contains a single IPv6 address that is represented as eight colon separated groups of four lowercase hexadecimal digits, each group representing 16 bits as specified in RFC 4291. Note: Shortened addresses, suppressed zeros, and embedded IPv4 addresses are not tested.*
Test Steps	• Configure the TOE with the correct reference identifier. CN-ID • Create a certificate containing a wildcard that is not in the left-most label of the CN. • Utilize acumen-tls tool to start a TLS server with certificate containing a wildcard that is not in the left-most label of the presented identifier and verify the connection was unsuccessful. • Verify via TOE logs that connection was unsuccessful. • Verify via packet capture that connection was unsuccessful.

	DNS-ID • Create a certificate containing a wildcard that is not in the left-most label of the SAN. • Utilize acumen-tls tool to start a TLS server with certificate containing a wildcard that is not in the left-most label of the presented identifier and verify the connection was unsuccessful. • Verify via TOE logs that connection was unsuccessful. • Verify via packet capture that connection was unsuccessful.
Pass/Fail with Explanation	Pass. TOE rejects TLS connections when the wildcard is not in the left-most label of the certificate CN or SAN field. This meets the testing requirement.

7.1.2.41 FCS_TLSC_EXT.1.2 TEST #5 (2)(A) (CPP_ND_V3.0E)

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 5 [conditional]: 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): ii. [conditional]: 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.)
Note	*Note that the following tests are marked conditional and are applicable under the following conditions:* *a) For TLS-based trusted channel communications according to FTP_ITC.1 where RFC 6125 is selected, tests 1-6 are applicable.* *or* *b) For TLS-based trusted path communications according to FTP_TRP where RFC 6125 is selected, tests 1-6 are applicable.* *or*

	c) *For TLS-based trusted path communications according to FPT_ITT.1 where RFC 6125 is selected, tests 1-6 are applicable. Where RFC 5280 is selected, only test 7 is applicable.* *Note that for some tests additional conditions apply.* *IP addresses are binary values that must be converted to a textual representation when presented in the CN of a certificate. When testing IP addresses in the CN, the evaluator shall follow the following formatting rules:* • *IPv4: The CN contains a single address that is represented a 32-bit numeric address (IPv4) is written in decimal as four numbers that range from 0-255 separated by periods as specified in RFC 3986.* • *IPv6: The CN contains a single IPv6 address that is represented as eight colon separated groups of four lowercase hexadecimal digits, each group representing 16 bits as specified in RFC 4291. Note: Shortened addresses, suppressed zeros, and embedded IPv4 addresses are not tested.*
Test Steps	• Configure the TOE with the correct reference identifier. CN-ID: • Create a certificate using a wildcard in the left-most label of CN. • Utilize acumen-tls tool to start a TLS server with certificate containing a wildcard in the left-most label of CN and verify the connection was successful. • Verify via packet capture that connection was successful. DNS-ID: • Create a certificate using a wildcard in the left-most label of SAN. • Utilize acumen-tls tool to start a TLS server with certificate containing a wildcard in the left-most label of SAN and verify the connection was successful. • Verify via packet capture that connection was successful.
Pass/Fail with Explanation	Pass. TOE accepts the server certificate with wildcards in the CN or SAN field. This meets the testing requirements.

7.1.2.42 FCS_TLSC_EXT.1.2 TEST #5 (2)(B) (CPP_ND_V3.0E)

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 5 [conditional]: 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): ii. [conditional]: 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.)
Note	*Note that the following tests are marked conditional and are applicable under the following conditions:* *a) For TLS-based trusted channel communications according to FTP_ITC.1 where RFC 6125 is selected, tests 1-6 are applicable.* *or* *b) For TLS-based trusted path communications according to FTP_TRP where RFC 6125 is selected, tests 1-6 are applicable.* *or* *c) For TLS-based trusted path communications according to FPT_ITT.1 where RFC 6125 is selected, tests 1-6 are applicable. Where RFC 5280 is selected, only test 7 is applicable.* *Note that for some tests additional conditions apply.* *IP addresses are binary values that must be converted to a textual representation when presented in the CN of a certificate. When testing IP addresses in the CN, the evaluator shall follow the following formatting rules:* • *IPv4: The CN contains a single address that is represented a 32-bit numeric address (IPv4) is written in decimal as four numbers that range from 0-255 separated by periods as specified in RFC 3986.* • *IPv6: The CN contains a single IPv6 address that is represented as eight colon separated groups of four lowercase hexadecimal digits, each group representing 16 bits as specified in RFC 4291. Note: Shortened addresses, suppressed zeros, and embedded IPv4 addresses are not tested.*
Test Steps	Note: Using same certificates from FCS_TLSC_EXT.1.2 TEST #5 (2)(A) • Configure the TOE with the correct reference identifier. <u>CN-ID:</u> • Utilize acumen-tls tool to start a TLS server with certificate containing a wildcard in the left-most label of CN and verify the connection was unsuccessful.

	• Verify via TOE logs that the connection was unsuccessful. • Verify via packet capture that the connection was unsuccessful. DNS-ID: • Utilize acumen-tls tool to start a TLS server with certificate containing a wildcard in the left-most label of SAN and verify the connection was unsuccessful. • Verify via TOE logs that the connection was unsuccessful. • Verify via packet capture that the connection was unsuccessful.
Pass/Fail with Explanation	Pass. The TOE rejects the TLS connection when the reference identifier is configured without a left-most label as in the certificate SAN or CN field. This meets the testing requirements.

7.1.2.43 FCS_TLSC_EXT.1.2 TEST #5 (2)(C) (CPP_ND_V3.0E)

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 5 [conditional]: 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): ii. [conditional]: 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.)
Note	*Note that the following tests are marked conditional and are applicable under the following conditions:* *a) For TLS-based trusted channel communications according to FTP_ITC.1 where RFC 6125 is selected, tests 1-6 are applicable.* *or* *b) For TLS-based trusted path communications according to FTP_TRP where RFC 6125 is selected, tests 1-6 are applicable.* *or* *c) For TLS-based trusted path communications according to FPT_ITT.1 where RFC 6125 is selected, tests 1-6 are applicable. Where RFC 5280 is selected, only test 7 is applicable.*

	Note that for some tests additional conditions apply. *IP addresses are binary values that must be converted to a textual representation when presented in the CN of a certificate. When testing IP addresses in the CN, the evaluator shall follow the following formatting rules:* • *IPv4: The CN contains a single address that is represented a 32-bit numeric address (IPv4) is written in decimal as four numbers that range from 0-255 separated by periods as specified in RFC 3986.* • *IPv6: The CN contains a single IPv6 address that is represented as eight colon separated groups of four lowercase hexadecimal digits, each group representing 16 bits as specified in RFC 4291. Note: Shortened addresses, suppressed zeros, and embedded IPv4 addresses are not tested.*
Test Steps	Note: Using same certificates from FCS_TLSC_EXT.1.2 TEST #5 (2)(A) • Configure the correct reference identifier into the TOE. CN-ID: • Utilize acumen-tls tool to start a TLS server with certificate containing a wildcard in the left-most label of CN and verify the connection was unsuccessful. • Verify via TOE logs that the connection was unsuccessful. • Verify via packet capture that the connection was unsuccessful. DNS-ID: • Utilize acumen-tls tool to start a TLS server with certificate containing a wildcard in the left-most label of SAN and verify the connection was unsuccessful. • Verify via TOE logs that the connection was unsuccessful. • Verify via packet capture that the connection was unsuccessful.
Pass/Fail with Explanation	Pass. The TOE rejects the TLS connection when the reference identifier is configured with two left-most labels. This meets the testing requirements.

7.1.2.44 FCS_TLSC_EXT.1.2 TEST #6 (CPP_ND_V3.0E)

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 6: Objective: The objective of this test is to ensure the TOE is able to differentiate between IP address identifiers that are not allowed to contain wildcards and other types of identifiers that may contain wildcards. [conditional] If IP address identifiers 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).
Note	*Note that the following tests are marked conditional and are applicable under the following conditions:* *a) For TLS-based trusted channel communications according to FTP_ITC.1 where RFC 6125 is selected, tests 1-6 are applicable.* *or* *b) For TLS-based trusted path communications according to FTP_TRP where RFC 6125 is selected, tests 1-6 are applicable.* *or* *c) For TLS-based trusted path communications according to FPT_ITT.1 where RFC 6125 is selected, tests 1-6 are applicable. Where RFC 5280 is selected, only test 7 is applicable.* *Note that for some tests additional conditions apply.* *IP addresses are binary values that must be converted to a textual representation when presented in the CN of a certificate. When testing IP addresses in the CN, the evaluator shall follow the following formatting rules:* • *IPv4: The CN contains a single address that is represented a 32-bit numeric address (IPv4) is written in decimal as four numbers that range from 0-255 separated by periods as specified in RFC 3986.* • *IPv6: The CN contains a single IPv6 address that is represented as eight colon separated groups of four lowercase hexadecimal digits, each group representing 16 bits as specified in RFC 4291. Note: Shortened addresses, suppressed zeros, and embedded IPv4 addresses are not tested.*
Test Steps	IPv4 • Configure the TOE with the correct reference identifier. • Create a certificate with a wildcard in CN and missing SAN. • Utilize acumen-tls tool to start a TLS server with certificate which has wildcard in CN and missing SAN and verify the connection was unsuccessful. • Verify via TOE logs that connection was unsuccessful. • Verify via packet capture that connection was unsuccessful.
Pass/Fail with Explanation	Pass. The TOE rejects the TLS connection when the server certificate contains a CN that matches the reference identifier, except one of the groups has been replaced with a wildcard and does not contain the SAN extension. This meets the testing requirements.

7.1.2.45 FCS_TLSC_EXT.1.2 TEST #7A (CPP_ND_V3.0E)

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 shall modify 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): i. 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.
Note	*Note that the following tests are marked conditional and are applicable under the following conditions:* *a) For TLS-based trusted channel communications according to FTP_ITC.1 where RFC 6125 is selected, tests 1-6 are applicable.* *or* *b) For TLS-based trusted path communications according to FTP_TRP where RFC 6125 is selected, tests 1-6 are applicable.* *or* *c) For TLS-based trusted path communications according to FPT_ITT.1 where RFC 6125 is selected, tests 1-6 are applicable. Where RFC 5280 is selected, only test 7 is applicable.* *Note that for some tests additional conditions apply.* *IP addresses are binary values that must be converted to a textual representation when presented in the CN of a certificate. When testing IP addresses in the CN, the evaluator shall follow the following formatting rules:* • *IPv4: The CN contains a single address that is represented a 32-bit numeric address (IPv4) is written in decimal as four numbers that range from 0-255 separated by periods as specified in RFC 3986.* • *IPv6: The CN contains a single IPv6 address that is represented as eight colon separated groups of four lowercase hexadecimal digits, each group representing 16 bits as specified in RFC 4291. Note: Shortened addresses, suppressed zeros, and embedded IPv4 addresses are not tested.*
Pass/Fail with Explanation	N/A. FPT_ITT is not claimed in ST.

7.1.2.46 FCS_TLSC_EXT.1.2 TEST #7B (CPP_ND_V3.0E)

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 shall modify each attribute type in accordance with the matching criteria described in the TSS (e.g. creating a mismatch of one attribute type at a time while other attribute types contain values that will match a portion of the reference identifier): The evaluator shall present a server certificate that contains a valid identifier as an attribute type other than the expected attribute type (e.g. if the TOE is configured to expect id-at-serialNumber=correct_identifier, the certificate could instead include id-at-name=correct_identifier), and does not contain the SAN extension. The evaluator shall verify that the connection fails. Remark: Some systems might require the presence of the SAN extension. In this case the connection would still fail but for the reason of the missing SAN extension instead of the mismatch of CN and reference identifier. Both reasons are acceptable to pass this test.
Note	*Note that the following tests are marked conditional and are applicable under the following conditions:* *a) For TLS-based trusted channel communications according to FTP_ITC.1 where RFC 6125 is selected, tests 1-6 are applicable.* *or* *b) For TLS-based trusted path communications according to FTP_TRP where RFC 6125 is selected, tests 1-6 are applicable.* *or* *c) For TLS-based trusted path communications according to FPT_ITT.1 where RFC 6125 is selected, tests 1-6 are applicable. Where RFC 5280 is selected, only test 7 is applicable.* *Note that for some tests additional conditions apply.* *IP addresses are binary values that must be converted to a textual representation when presented in the CN of a certificate. When testing IP addresses in the CN, the evaluator shall follow the following formatting rules:*

	• *IPv4: The CN contains a single address that is represented a 32-bit numeric address (IPv4) is written in decimal as four numbers that range from 0-255 separated by periods as specified in RFC 3986.* • *IPv6: The CN contains a single IPv6 address that is represented as eight colon separated groups of four lowercase hexadecimal digits, each group representing 16 bits as specified in RFC 4291. Note: Shortened addresses, suppressed zeros, and embedded IPv4 addresses are not tested.*
Pass/Fail with Explanation	N/A. FPT_ITT is not claimed in ST.

7.1.2.47 FCS_TLSC_EXT.1.2 TEST #7C (CPP_ND_V3.0E)

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 shall modify 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): ii. 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.
Note	*Note that the following tests are marked conditional and are applicable under the following conditions:* *a) For TLS-based trusted channel communications according to FTP_ITC.1 where RFC 6125 is selected, tests 1-6 are applicable.* *or* *b) For TLS-based trusted path communications according to FTP_TRP where RFC 6125 is selected, tests 1-6 are applicable.* *or* *c) For TLS-based trusted path communications according to FPT_ITT.1 where RFC 6125 is selected, tests 1-6 are applicable. Where RFC 5280 is selected, only test 7 is applicable.* *Note that for some tests additional conditions apply.*

	IP addresses are binary values that must be converted to a textual representation when presented in the CN of a certificate. When testing IP addresses in the CN, the evaluator shall follow the following formatting rules: • *IPv4: The CN contains a single address that is represented a 32-bit numeric address (IPv4) is written in decimal as four numbers that range from 0-255 separated by periods as specified in RFC 3986.* • *IPv6: The CN contains a single IPv6 address that is represented as eight colon separated groups of four lowercase hexadecimal digits, each group representing 16 bits as specified in RFC 4291. Note: Shortened addresses, suppressed zeros, and embedded IPv4 addresses are not tested.*
Pass/Fail with Explanation	N/A. FPT_ITT is not claimed in ST.

7.1.2.48 FCS_TLSC_EXT.1.2 TEST #7D (CPP_ND_V3.0E)

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 shall modify 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): iii. The evaluator shall confirm that all use of wildcards results in connection failure regardless of whether the wildcards are used in the left or right side of the presented identifier. (Remark: Use of wildcards is not addressed within RFC 5280.)
	Note that the following tests are marked conditional and are applicable under the following conditions: *a) For TLS-based trusted channel communications according to FTP_ITC.1 where RFC 6125 is selected, tests 1-6 are applicable.* *or* *b) For TLS-based trusted path communications according to FTP_TRP where RFC 6125 is selected, tests 1-6 are applicable.* *or* *c) For TLS-based trusted path communications according to FPT_ITT.1 where RFC 6125 is selected, tests 1-6 are applicable. Where RFC 5280 is selected, only test 7 is applicable.* *Note that for some tests additional conditions apply.*

	IP addresses are binary values that must be converted to a textual representation when presented in the CN of a certificate. When testing IP addresses in the CN, the evaluator shall follow the following formatting rules: • *IPv4: The CN contains a single address that is represented a 32-bit numeric address (IPv4) is written in decimal as four numbers that range from 0-255 separated by periods as specified in RFC 3986.* • *IPv6: The CN contains a single IPv6 address that is represented as eight colon separated groups of four lowercase hexadecimal digits, each group representing 16 bits as specified in RFC 4291. Note: Shortened addresses, suppressed zeros, and embedded IPv4 addresses are not tested.*
Pass/Fail with Explanation	N/A. FPT_ITT is not claimed in ST.

7.1.2.49 FCS_TLSC_EXT.1.3 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall demonstrate that using an invalid certificate results in the function failing as follows: Test 1: 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. This test is covered as a part of FIA_X509_EXT.1.1 Test #1a. This meets the testing requirements.

7.1.2.50 FCS_TLSC_EXT.1.3 TEST #2 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall demonstrate that using an invalid certificate results in the function failing as follows: Test 2 [conditional]: If "except with the following administrator override" is selected, the evaluator shall change the presented certificate(s) or modify the operational environment, so that certificate validation fails due to the TSF's inability to determine revocation status. The evaluator shall verify that the certificate is not accepted by the TSF until the Security Administrator authorizes

	the TSF to establish the connection and this action results in the Trusted Channel being successfully established.
Pass/Fail with Explanation	N/A. “without any administrator override mechanism” selection is done in ST. This meets the testing requirements.

7.1.2.51 FCS_TLSC_EXT.1.3 TEST #3 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall demonstrate that using an invalid certificate results in the function failing as follows: Test 3: While performing testing of invalid TLS Client Reference Identifiers, expired X.509 certificates, and invalid X.509 trust chains; the evaluator shall ensure the TSF does not present an administrator override option, with the expcetion of failure to determine revocation status (if selected). Note: This should be a review of behiavor observed while performing other tests.
Test Steps	Failed matching of the reference identifier The requirement for this test is covered in FCS_TLSC_EXT.1.2 Test #2. Failed validation of the certificate path The requirement for this test is covered in FIA_X509_EXT.1.1/Rev Test #1b. Failed validation of the expiration date The requirement for this test is covered in FIA_X509_EXT.1.1/Rev Test #2. Failed determination of the revocation status The requirement for this test is covered in FIA_X509_EXT.2 Test #1.
Pass/Fail with Explanation	Pass. TOE rejects TLS connection for server certificates using invalid TLS Client Reference Identifiers, expired X.509 certificates, and invalid X.509 trust chains, which is satisfied by the respective referenced test cases. This meets the testing requirements.

7.1.2.52 FCS_TLSC_EXT.1.4 TEST #1 (CPP_ND_V3.0E)

Item	Data

Test Assurance Activity	The evaluator shall perform the following tests: Test 1 [conditional]: If “not present the Supported Groups Extension” is selected, the evaluator shall examine the Client Hello message and verify it does not contain the Supported Groups extension.
Pass/Fail with Explanation	N/A. “Present the Supported Groups Extension” is selected in ST

7.1.2.53 FCS_TLSC_EXT.1.4 TEST #2 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 2 [conditional]: If “present the Supported Groups Extension” is selected, the evaluator shall configure the server to perform ECDHE or DHE (as applicable) key exchange using each of the TOE’s supported groups. The evaluator shall verify that the connection succeeds. This test shall be repeated for each type of key exchange message/extension supported (i.e. Key Share extension for TLS 1.3 and Server Key Exchange Message for TLS 1.2).
Test Steps	• Utilize acumen-tls tool to start a TLS server using secp256r1 curve and verify the connection was successful. • Verify via TOE logs the connection was successful. • Verify via packet capture the connection was successful • Utilize acumen-tls tool to start a TLS server using secp384r1 curve and verify the connection was successful. • Verify via TOE logs the connection was successful. • Verify via packet capture the connection was successful. • Utilize acumen-tls tool to start a TLS server using secp521r1 curve and verify the connection was successful. • Verify via TOE logs the connection was successful. • Verify via packet capture the connection was successful.
Pass/Fail with Explanation	Pass. TOE is able to establish a TLS connection using each of the claimed EC groups. This meets the testing requirements.

7.1.2.54 FCS_TLSC_EXT.1.4 TEST #3 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests:

	Test 3 [conditional]: If secp curves are selected, the evaluator shall configure the server to perform an ECDHE key exchange in the TLS connection using a non-supported curve and shall verify that the connection fails and no application data flows. The non-supported curve shall be as similar to the selected curve(s) as possible (i.e. a non-selected curve when not all curves are selected or P-224). This test shall be repeated for each type of key exchange message/extension supported (i.e. Key Share extension for TLS 1.3 and Server Key Exchange Message for TLS 1.2).
Test Steps	• Utilize Acumen-tls tool to start a TLS server to send an unsupported curve. • Verify via TOE logs that the connection was unsuccessful. • Verify via packet capture that the connection was unsuccessful.
Pass/Fail with Explanation	Pass. The TOE rejects TLS connections when an unsupported curve is used to perform ECDHE key exchange. This meets the testing requirements.

7.1.2.55 FCS_TLSC_EXT.1.4 TEST #4A (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 4a [conditional, for TLS 1.3 only]: If ffdhe curves are selected, the evaluator shall configure the server to perform a DHE key exchange in the TLS connection using a non-supported group and shall verify that the connection fails and no application data flows. The non-supported group shall be as similar to the selected group(s) as possible (i.e. a non-selected group when not all groups are selected or undefined Codepoint 0x0105 (ffdhe8192 + 1)).
Pass/Fail with Explanation	N/A. TOE does not support ffdhe groups.

7.1.2.56 FCS_TLSC_EXT.1.4 TEST #4B (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 4b [conditional, for TLS 1.2 only]: If ffdhe curves are selected, the evaluator shall configure the server to return DHE parameters in the Server Key Exchange in the TLS connection that do not meet the construction for any claimed ffdhe group. The evaluator shall verify that the connection fails and no application data flows. If the TOE client supports any server-returned DHE parameter set, then this test is not applicable.

Pass/Fail with Explanation	N/A. TOE does not support ffdhe groups.

7.1.2.57 FCS_TLSC_EXT.1.5 TEST #1A (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 1 [conditional]: The evaluator shall perform the following tests if "present the signature_algorithms extension" is selected: i. The evaluator shall examine the Client Hello message and verify it contains the signature_algorithms extension and the SignatureSchemes match the SignatureSchemes specified in the requirement.
Test Steps	• Utilize acumen-tls tool to start a TLS server and verify the client hello contains only the signature_algorithms extension and SignatureSchemes. • Verify via packet capture that the client hello contains only the signature_algorithms extension and SignatureSchemes.
Pass/Fail with Explanation	Pass. The TOE's Client Hello message field contains the signature_algorithms extension and only claimed SignatureSchemes. This meets the testing requirements.

7.1.2.58 FCS_TLSC_EXT.1.5 TEST #1B (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 1 [conditional]: The evaluator shall perform the following tests if "present the signature_algorithms extension" is selected: ii. The evaluator shall establish a TLS connection using each of the SignatureSchemes specified by the requirement and observes the session is successfully completed. The evaluator shall ensure the test server sends a leaf Certificate that has a public key algorithm that is consistent with the SignatureScheme being tested. **For TLS 1.2 and if the ciphersuite is DHE or ECDHE**, the evaluator shall ensure that the server sends Server Key Exchange messages consistent with the SignatureScheme being tested. **For TLS 1.3**, the evaluator shall ensure that the server sends Certificate Verify messages consistent with the SignatureScheme being tested.
Test Steps	• Utilize acumen-tls tool to start a TLS server with certificate using SignatureScheme of ecdsa_secp256r1 with sha256 and verify the connection was successful.

	• Verify that the TLS session was established using the ecdsa_secp256r1 with sha256 SignatureScheme via packet capture. • Utilize acumen-tls tool to start a TLS server with certificate using SignatureScheme of ecdsa_secp384r1 with sha384 and verify the connection was successful. • Verify that the TLS session was established using the ecdsa_secp384r1 with sha384 SignatureScheme via packet capture. • Utilize acumen-tls tool to start a TLS server with certificate using SignatureScheme of ecdsa_secp521r1 with sha521 and verify the connection was successful. • Verify that the TLS session was established using the ecdsa_secp521r1 with sha521 SignatureScheme via packet capture.
Pass/Fail with Explanation	Pass. TOE is able to establish a TLS connection using each of the claimed SignatureScheme. Support for TLS version 1.3 is not claimed by ST. This meets the requirements.

7.1.2.59 FCS_TLSC_EXT.1.5 TEST #2A (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 2 [conditional]: The evaluator shall perform the following tests if “present the signature_algorithms_cert extension” is selected: i. The evaluator shall examine the Client Hello message and verify it contains the signature_algorithms_cert extension and the SignatureSchemes match the SignatureSchemes specified in the requirement.
Pass/Fail with Explanation	N/A. “present the signature_algorithms_cert extension” not selected in ST.

7.1.2.60 FCS_TLSC_EXT.1.5 TEST #2B (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 2 [conditional]: The evaluator shall perform the following tests if “present the signature_algorithms_cert extension” is selected: ii. The evaluator shall establish a TLS connection using a certificate chain using each of the SignatureSchemes specified by the requirement. The

	evaluator shall ensure the signatures used in the certificate chain are consistent with the SignatureScheme being tested.
Pass/Fail with Explanation	N/A. “present the signature_algorithms_cert extension” not selected in ST.

7.1.2.61 FCS_TLSC_EXT.1.6 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	[conditional]: If the TSF provides the ability of configuring the list of supported ciphersuites, the evaluator shall establish a TLS connection using one of the possible configurations of the list of supported ciphersuites. The evaluator shall then change the configuration and repeat the test. The evaluator shall verify that the behavior of the TOE has changed according to the modification of the list of ciphers. This test shall be repeated for all supported TLS versions. If the TSF does not provide the ability of configuring the list of supported ciphersuites, this test shall be omitted.
Test Steps	• Configure on TOE the list of supported cipher suite mentioned in the ST. • Verify the configuration via logs. TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA • Utilize acumen-tls tool to start a TLS server using TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA ciphersuite. • Verify that the TLS session was encrypted using TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA ciphersuite via packet capture. TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA • Utilize acumen-tls tool to start a TLS server using TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA ciphersuite. • Verify that the TLS session was encrypted using TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA ciphersuite via packet capture. TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA 256 • Utilize acumen-tls tool to start a TLS server using TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA 256 ciphersuite. • Verify that the TLS session was encrypted using TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA 256 ciphersuite via packet capture. TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA 384 • Utilize acumen-tls tool to start a TLS server using TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA 384 ciphersuite.

	• Verify that the TLS session was encrypted using TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA 384 ciphersuite via packet capture. TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 • Utilize acumen-tls tool to start a TLS server using TLS_ECDHE_ ECDSA _WITH_AES_128_GCM_SHA256. • Verify that the TLS session was encrypted using TLS_ECDHE_ ECDSA _WITH_AES_128_GCM_SHA256 ciphersuite via packet capture. TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 • Utilize acumen-tls tool to start a TLS server using TLS_ECDHE_ ECDSA _WITH_AES_256_GCM_SHA384 ciphersu • Verify that the TLS session was encrypted using TLS_ECDHE_ ECDSA _WITH_AES_256_GCM_SHA384 ciphersuite via packet capture.
Pass/Fail with Explanation	Pass. The TOE can configure each claimed TLS ciphersuites and also establish a TLS connection using each of the claimed ciphersuites. This meets the testing requirements.

7.1.2.62 FCS_TLSC_EXT.1.7 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall establish a TLS connection with a server and observe that the early data extension and the post-handshake client authentication extension according to RFC 8446 Section 4.2 are not advertised in the Client Hello Message. This test shall be executed for all TLS versions supported by the TOE.
Test Steps	• Attempt a TLS connection from the TOE to the server and ensure it was successful. • Verify via packet capture that Client Hello does not advertise the early data extension and the post-handshake client authentication extension according to RFC 8446 Section 4.2.
Pass/Fail with Explanation	Pass. TOE Client Hello message does not advertise the early data extension and the post-handshake client authentication extension according to RFC 8446 Section 4.2. This meets the testing requirements.

7.1.2.63 FCS_TLSC_EXT.1.8 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	No test assurance activities have been identified.

Pass/Fail with Explanation	Pass. No test assurance activities have been identified.

7.1.2.64 FCS_TLSC_EXT.1.9 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 1 [conditional]: If "support TLS 1.2 secure renegotiation…" is selected, the evaluator shall use a network packet analyzer/sniffer to capture a TLS 1.2 handshake between the two TLS endpoints. The evaluator shall verify that either the “renegotiation_info” field or the SCSV ciphersuite is included in the ClientHello message during the initial handshake.
Pass/Fail with Explanation	N/A. The TOE reject TLS 1.2 secure renegotiation.

7.1.2.65 FCS_TLSC_EXT.1.9 TEST #2 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 2 [conditional]: If "support TLS 1.2 secure renegotiation…" is selected, the evaluator shall perform a TLS 1.2 handshake and verify the TOE TLS Client’s handling of ServerHello messages received during the initial handshake that include the “renegotiation_info” extension. The evaluator shall modify the length portion of this field in the ServerHello message to be non-zero and verify that the TOE TLS client sends a failure and terminates the connection. The evaluator shall verify that a properly formatted field results in a successful TLS connection.
Pass/Fail with Explanation	N/A. The TOE rejects TLS 1.2 secure renegotiation.

7.1.2.66 FCS_TLSC_EXT.1.9 TEST #3 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 3 [conditional]: If "support TLS 1.2 secure renegotiation…" is selected, the evaluator shall perform a TLS 1.2 handshake and verify that ServerHello

	messages received during secure renegotiation contain the “renegotiation_info” extension. The evaluator shall modify either the “client_verify_data” or “server_verify_data” value and verify that the TOE TLS client terminates the connection.
Pass/Fail with Explanation	N/A. The TOE rejects TLS 1.2 secure renegotiation.

7.1.2.67 FCS_TLSC_EXT.1.9 TEST #4A **[TD0899]** (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 4a [conditional, if the TOE supports TLS 1.3]: The evaluator shall initiate a TLS session between the TSF and a test TLS 1.3 server that completes a compliant TLS 1.3 handshake, followed by a hello request message. The evaluator shall observe that the TSF completes the initial TLS 1.3 handshake successfully, but terminates the session on receiving the hello request message. It is preferred that the TSF sends a fatal error alert message (e.g., unexpected message) in response to this, but it is acceptable that the TSF terminates the connection silently (i.e., without sending a fatal error alert). **TD0899 has been applied**.
Pass/Fail with Explanation	N/A. Support for TLS version 1.3 is not claimed by ST. This meets the testing requirements.

7.1.2.68 FCS_TLSC_EXT.1.9 TEST #4B **[TD0899]** (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 4b [conditional, if the TOE supports TLS 1.2 and rejects TLS 1.2 renegotiation attempts]: The evaluator shall initiate a TLS session between the so-configured TSF and a test TLS 1.2 server that is configured to perform a compliant handshake, followed by a hello request. The evaluator shall confirm that the TSF completes the initial handshake successfully but does not initiate renegotiation after receiving the hello request. **TD0899 has been applied**.

Test Steps	• Utilize acumen-tls-tool to perform a compliant handshake with the TOE. • Verify via packet capture that the TOE rejects the hello request packet. • Verify via TOE logs the connection was terminated.
Pass/Fail with Explanation	Pass. The TOE rejects the TLS session renegotiation attempt with the TLS server. This meets the testing requirements.

7.1.2.69 FCS_TLSS_EXT.1.1 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 1: The evaluator shall establish a TLS connection using each of the ciphersuites specified by the requirement. This connection may be established as part of the establishment of a higher-level protocol, e.g., as part of an HTTPS session. It is sufficient to observe the successful negotiation of a ciphersuite to satisfy the intent of the test; it is not necessary to examine the characteristics of the encrypted traffic to discern the ciphersuite being used (for example, that the cryptographic algorithm is 128-bit AES and not 256-bit AES).
Test Steps	• Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA. • Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA. • Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA 256 • Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA_384. • Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256. • Verify the required ciphersuite with packet capture. • Establish a connection with the TOE over TLS using the ciphersuite TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384. • Verify the required ciphersuite with packet capture.

Pass/Fail with Explanation	Pass. The TOE can establish a successful connection via the supported ciphersuites. This meets the testing requirements.

7.1.2.70 FCS_TLSS_EXT.1.1 TEST #2A (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 2: The evaluator shall perform the following tests: i. The evaluator shall send a Client Hello to the server with a list of ciphersuites that does not contain any of the ciphersuites in the server's ST and verify that the server denies the connection.
Test Steps	• Using the acumen-tls tool as a client, attempt to establish a TLS connection to the TOE using an unsupported cipher suite in the Client Hello and verify the connection failure. • Verify the failure logs on TOE showing failure due to no shared cipher. • Verify the connection fails via packet capture.
Pass/Fail with Explanation	Pass. The TOE rejects TLS connections with the unsupported ciphersuites. This meets the testing requirement.

7.1.2.71 FCS_TLSS_EXT.1.1 TEST #2B (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 2: The evaluator shall perform the following tests: ii. [conditional]: Perform this test only if support of TLS 1.2 is claimed. The evaluator shall send a Client Hello to the server containing only the TLS_NULL_WITH_NULL_NULL ciphersuite and verify that the server denies the connection.
Test Steps	• Using the acumen-tls tool as a client, attempt to establish a TLS connection to the TOE using TLS_NULL_WITH_NULL_NULL cipher suite in the Client Hello and verify the connection failure. • Verify the failure logs on TOE showing failure due to no shared cipher. • Verify the connection fails via packet capture.
Pass/Fail with Explanation	Pass. The TOE rejects TLS connections with the unsupported TLS_NULL_WITH_NULL_NULL ciphersuites. This meets the testing requirement.

7.1.2.72 FCS_TLSS_EXT.1.1 TEST #3A (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 3: The evaluator shall perform the following modifications to the traffic: i. [conditional]: Perform this test only if support of TLS 1.2 is claimed. Modify a byte in the Client Finished handshake message, and verify that the server rejects the connection and does not send any application data.
Test Steps	• Run the acumen-tlss tool as a client with a modified client finished message and wait for the connection, the connection should fail. • Verify the failure logs on the device, ensuring they indicate the failure occurred due to a decryption failure. • Verify the unsuccessful connection via packet capture.
Pass/Fail with Explanation	Pass. The TOE rejects the connection after receiving the modified Client Handshake message. This meets the test requirements.

7.1.2.73 FCS_TLSS_EXT.1.1 TEST #3B (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 3: The evaluator shall perform the following modifications to the traffic: ii. (The intent of this test is to ensure that the server's TLS implementation immediately makes use of the key exchange and authentication algorithms to: a) Correctly encrypt TLS Finished message and b) Encrypt every TLS message after session keys are negotiated.) [conditional]: Perform this test only if support of TLS 1.2 is claimed. The evaluator shall use one of the claimed ciphersuites to complete a successful handshake and observe transmission of properly encrypted application data. The evaluator shall verify that no Alert with alert level Fatal (2) messages were sent. The evaluator shall verify that the Finished message (Content type hexadecimal 16 and handshake message type hexadecimal 14) is sent immediately after the server's ChangeCipherSpec (Content type hexadecimal 14) message. The evaluator shall examine the Finished message (encrypted example in hexadecimal of a TLS record containing a Finished message, 16 03 03 00 40 11 22 33 44 55…) and confirm that it does not contain unencrypted data (unencrypted example in hexadecimal of a TLS record containing a Finished message, 16 03 03 00

	40 14 00 00 0c…), by verifying that the first byte of the encrypted Finished message does not equal hexadecimal 14 for at least one of three test messages. There is a chance that an encrypted Finished message contains a hexadecimal value of '14' at the position where a plaintext Finished message would contain the message type code '14'. If the observed Finished message contains a hexadecimal value of '14' at the position where the plaintext Finished message would contain the message type code, the test shall be repeated three times in total. In case the value of '14' can be observed in all three tests it can be assumed that the Finished message has indeed been sent in plaintext and the test has to be regarded as 'failed'. Otherwise it has to be assumed that the observation of the value '14' has been due to chance and that the Finished message has indeed been sent encrypted. In that latter case the test shall be regarded as 'passed'.
Test Steps	• Initiate a connection to the TOE with acumen-tls tool as a client. • Verify that no Alert with alert level Fatal (2) messages were sent. • Verify that the Finished message (Content type hexadecimal 16 and handshake message type hexadecimal 14) is sent immediately after the server's ChangeCipherSpec (Content type hexadecimal 14) message. • Examine the Finished message and confirm that it does not contain unencrypted data by verifying that the first byte of the encrypted Finished message does not equal hexadecimal 14.
Pass/Fail with Explanation	Pass. The Finished message contains Hexadecimal 16 and is sent immediately after Hexadecimal 14 in the ChangeCipherSpec message. The first byte of the encrypted Finished message does not equal hexadecimal 14. This meets the testing requirement.

7.1.2.74 FCS_TLSS_EXT.1.1 TEST #3C (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 3: The evaluator shall perform the following modifications to the traffic: iii. [conditional]: Perform this test only if support of TLS 1.3 is claimed. The evaluator shall use a client to send a Client Hello message containing a single curve in the Supported Groups extension. The curve that is selected to be presented in this extension should not be supported by the TOE. The evaluator shall verify that the TOE disconnects after receiving the Client Hello message.
Pass/Fail with Explanation	N/A. Support for TLS version 1.3 is not claimed by ST. This meets the testing requirements.

7.1.2.75 FCS_TLSS_EXT.1.1 TEST #3D (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 3: The evaluator shall perform the following modifications to the traffic: iv. [conditional]: Perform this test only if support of TLS 1.3 is claimed. The evaluator shall use a client to send a Client Hello message containing multiple curves in the Supported Groups extension. These curves should be chosen such that only one of these curves is supported by the TOE. The evaluator shall verify that the TOE responds with a Hello Retry Request message selecting the supported curve. This shall be reflected in the Key Share extension of the Hello Retry Request message.
Pass/Fail with Explanation	N/A. Support for TLS version 1.3 is not claimed by ST. This meets the testing requirements.

7.1.2.76 FCS_TLSS_EXT.1.1 TEST #4 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 4: The evaluator shall attempt to establish a TLS/SSL connection using each of the supported TLS/SSL versions (i.e., TLS 1.3, TLS 1.2, TLS 1.1, TLS 1.0, SSL 3.0, SSL 2.0). The client shall be configured so it only supports the version being tested. The evaluator shall verify that the versions specified in FCS_TLSS_EXT.1.1 are successfully established and all other versions not successfully established. If the TOE attempts to downgrade the version, it is acceptable for the test client to terminate the connection; however, the version selected by the TOE shall always be a version specified in FCS_TLSS_EXT.1.1.
Test Steps	• Use the acumen-tls tool as a client to initiate a connection to the TOE and verify that the connection fails for all non-supported SSL and TLS versions and passes for supported versions. SSL v2.0 • Verify the connection failed for SSL v2.0 with the logs. • Verify that the SSL v2.0 connection was denied using a packet capture. SSL v3.0 • Verify the connection failed for SSL v3.0 with the logs. • Verify that the SSL v3.0 connection was denied using a packet capture. TLS v1.0

	• Verify the connection failed for TLS1.0 with the logs. • Verify that the TLS v1.0 connection was denied using a packet capture. TLS v1.1 • Verify the connection failed for TLS1.1 with the logs. • Verify that the TLS v1.1 connection was denied using a packet capture. TLS v1.2 • Verify that the TLS v1.2 connection was successful using a packet capture. TLS v1.3 • Use the acumen-tls tool as a client to initiate a connection to the TOE with TLS 1.3 • Verify via TOE logs that the connection was unsuccessful. • Verify via Packet capture that the connection was unsuccessful.
Pass/Fail with Explanation	Pass. The TOE rejects all SSLv2, SSLv3, TLS v1.0, TLS v1.1 and TLS v1.3 connection attempts and accepts the connection when TLS 1.2 is used. This meets the testing requirement.

7.1.2.77 FCS_TLSS_EXT.1.2 & 1.3 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 1 [conditional]: If ECDHE ciphersuites/group are supported: The evaluator shall repeat this test for each supported elliptic curve. The evaluator shall attempt a connection using a supported ECDHE ciphersuite (TLS 1.2) or group (TLS 1.3) and a single supported elliptic curve specified in the supported groups extension. The Evaluator shall verify (through a packet capture or instrumented client) that the TOE selects the same curve in the Server Key Exchange (TLS 1.2) or Server Hello (key_share, for TLS 1.3) message and successfully establishes the connection. **For TLS 1.2**, the evaluator shall attempt a connection using a supported ECDHE ciphersuite and a single unsupported elliptic curve (e.g., secp192r1 (0x13)) specified in RFC 4492, Section 5.1.1. The evaluator shall verify that the TOE does not send a Server Hello message and the connection was not successfully established. **For TLS 1.3**, the evaluator shall attempt a connection using a supported ciphersuite and a single unsupported group. Both the key_share and supported_groups extensions must be set to the same unsupported group. The

	evaluator shall verify that the TOE does not send a Server Hello message and the connection was not successfully established.
Test Steps	Part1: • Run the openssl tool as a client, connect to the TOE using secp256r1 and verify that it was successful. • Verify with packet capture that the connection was established using the curve secp256r1. Part2: • Run the openssl tool as a client, establish a connection to TOE over TLS using the supported ECDHE ciphersuite and unsupported elliptical curve and verify the connection fails. • Verify the log on the device showing failure due to no shared cipher. • Verify the packet capture showing connection failure.
Pass/Fail with Explanation	Pass. The TOE rejects a TLS connection from the client when an unsupported elliptic curve is used and accepts a TLS connection from the client when a supported elliptic curve is used. Also, the TOE does not support TLSv1.3. This meets the testing requirement

7.1.2.78 FCS_TLSS_EXT.1.2 & 1.3 TEST #2A (CPP_ND_V3.0E) **[TD0973]**

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 2a [conditional]: If the TOE supports TLS 1.2 and DHE ciphersuites, the evaluator shall repeat the following test for each supported parameter size. If any configuration is necessary, the evaluator shall configure the TOE to use each supported Diffie-Hellman parameter size. The evaluator shall attempt a connection using a supported DHE ciphersuite. The evaluator shall verify (through a packet capture or instrumented client) that the TOE sends a Server Key Exchange Message where p Length is consistent with the ones configured Diffie-Hellman parameter size(s). **TD0973 has been applied.**
Pass/Fail with Explanation	N/A. This test is not applicable since DHE ciphersuites are not claimed in the ST.

7.1.2.79 FCS_TLSS_EXT.1.2 & 1.3 TEST #2B (CPP_ND_V3.0E) **[TD0973]**

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 2b: [conditional] If the TOE supports TLS 1.3 and 'ffdhe' parameters are selected, the evaluator shall repeat the following test for each supported FFDHE parameter size. If any configuration is necessary, the evaluator shall configure the TOE to use each supported Finite Field Diffie-Hellman parameter size. The evaluator shall verify (through a packet capture or instrumented client) that the TOE sends a Server Key Share Extension Message where the KeyShareServerHello structure contains a KeyShareEntry structure with an opaque key_exchange value whose Length is consistent with the configured Finite Field Diffie-Hellman parameter size(s). **TD0973 has been applied.**
Pass/Fail with Explanation	N/A. This test is not applicable since TLS version 1.3 is not claimed in the ST.

7.1.2.80 FCS_TLSS_EXT.1.2 & 1.3 TEST #3 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 3 [conditional]: If RSA key establishment ciphersuites are supported, the evaluator shall repeat this test for each RSA key establishment key size. If any configuration is necessary, the evaluator shall configure the TOE to perform RSA key establishment using a supported key size (e.g. by loading a certificate with the appropriate key size). The evaluator shall attempt a connection using a supported RSA key establishment ciphersuite. The evaluator shall verify (through a packet capture or instrumented client) that the TOE sends a certificate whose modulus is consistent with the configured RSA key size.
Pass/Fail with Explanation	N/A. The TOE does not support RSA Key establishment.

7.1.2.81 FCS_TLSS_EXT.1.4 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	*Test Objective: To demonstrate that the TOE will not resume a session for which the client failed to complete the handshake (independent of TOE support for session resumption).* The evaluator shall perform the following tests:

	Test 1 [conditional]: If the TOE does not support session resumption based on session IDs according to RFC 5246 (TLS 1.2) or session tickets according to RFC 5077 (TLS 1.2) or session resumption according to RFC 8446 (TLS 1.3), the evaluator shall perform the following test: i. For all supported TLS versions the client shall send a Client Hello with a zero-length session identifier and with a SessionTicket extension containing a zero-length ticket. A non-zero length session identifier for TLS 1.3 would result in testing compatibility mode which is not the objective of this test. For TLS 1.3, the evaluator shall ensure that a 'psk_key_exchange_modes' extension is included in the Client Hello. ii. The client verifies the server does not send a NewSessionTicket handshake message (at any point in the handshake). iii. The client verifies the Server Hello message contains a zerolength session identifier. For TLS 1.2 the client could alternatively pass the following steps (not applicable for TLS 1.3): Note: The following steps are only performed if the ServerHello message contains a non-zero length SessionID. iv. The client completes the TLS handshake and captures the SessionID from the ServerHello. v. The client sends a ClientHello containing the SessionID captured in step d). This can be done by keeping the TLS session in step d) open or start a new TLS session using the SessionID captured in step d). vi. The client verifies the TOE (1) implicitly rejects the SessionID by sending a ServerHello containing a different SessionID and by performing a full handshake (as shown in Figure 1 of RFC 4346 or RFC 5246), or (2) terminates the connection in some way that prevents the flow of application data. Remark: If multiple contexts are supported for session resumption, the session ID or session ticket may be obtained in one context for resumption in another context. It is possible that one or more contexts may only permit the construction of sessions to be reused in other contexts but not actually permit resumption themselves. For contexts which do not permit resumption, the evaluator is required to verify this behaviour subject to the description provided in the TSS. It is not mandated that the session establishment and session resumption share context. For example, it is acceptable for a control channel to establish and application channel to resume the session.
Test Steps	• Run the acumen-tls tool as a client to establish a TLS connection to the TOE. • Verify through packet capture that the client sends a client hello message with a zero-length session identifier and a session ticket extension containing a zero-length ticket.

	• Verify through packet capture that the server does not send a new session ticket handshake message at any point during the handshake and that the server hello message contains a zero-length session identifier.
Pass/Fail with Explanation	Pass. For TLS 1.2, the Client Hello message includes a zero-length session identifier and a SessionTicket extension containing a zero-length ticket. The server does not send a NewSessionTicket handshake message at any point during the handshake. Support for TLS version 1.3 is not claimed by ST. This meets the testing requirement.

7.1.2.82 FCS_TLSS_EXT.1.4 TEST #2A (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	*Test Objective: To demonstrate that the TOE will not resume a session for which the client failed to complete the handshake (independent of TOE support for session resumption).* Test 2 [conditional]: If the TOE supports session resumption using session IDs according to RFC 5246 (TLS 1.2), the evaluator shall carry out the following steps (note that for each of these tests, it is not necessary to perform the test case for each supported version of TLS): The evaluator shall conduct a successful handshake and capture the TOE-generated session ID in the Server Hello message. The evaluator shall then initiate a new TLS connection and send the previously captured session ID to show that the TOE resumed the previous session by responding with ServerHello containing the same SessionID immediately followed by ChangeCipherSpec and Finished messages (as shown in Figure 2 of RFC 4346 or RFC 5246). When the session is resumed, the evaluator shall verify on the TLS Client used for performing this test, that the TOE (TLS Server) has not advertised support for the early data extension.
Pass/Fail with Explanation	N/A. This test is not applicable since TOE does not support session resumption using session IDs according to RFC 5246 (TLS 1.2).

7.1.2.83 FCS_TLSS_EXT.1.4 TEST #2B (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	*Test Objective: To demonstrate that the TOE will not resume a session for which the client failed to complete the handshake (independent of TOE support for session resumption).* Test 2 [conditional]: If the TOE supports session resumption using session IDs according to RFC 5246 (TLS 1.2), the evaluator shall carry out the following steps

	(note that for each of these tests, it is not necessary to perform the test case for each supported version of TLS): i. The evaluator shall initiate a handshake and capture the TOE-generated session ID in the Server Hello message. The evaluator shall then, within the same handshake, generate or force an unencrypted fatal Alert message immediately before the client would otherwise send its ChangeCipherSpec message thereby disrupting the handshake. The evaluator shall then initiate a new Client Hello using the previously captured session ID, and verify that the server (1) implicitly rejects the session ID by sending a ServerHello containing a different SessionID and performing a full handshake (as shown in figure 1 of RFC 4346 or RFC 5246), or (2) terminates the connection in some way that prevents the flow of application data. Remark: If multiple contexts are supported for session resumption, for each of the above test cases, the session ID may be obtained in one context for resumption in another context. There is no requirement that the session ID be obtained and replayed within the same context subject to the description provided in the TSS. All contexts that can reuse a session ID constructed in another context must be tested. It is not mandated that the session establishment and session resumption share context. For example, it is acceptable for a control channel to establish and application channel to resume the session.
Pass/Fail with Explanation	N/A. This test is not applicable since TOE does not support session resumption using session IDs according to RFC 5246 (TLS 1.2).

7.1.2.84 FCS_TLSS_EXT.1.4 TEST #3A (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	*Test Objective: To demonstrate that the TOE will not resume a session for which the client failed to complete the handshake (independent of TOE support for session resumption).* Test 3 [conditional]: If the TOE supports session tickets according to RFC 5077 (supported only by TLS 1.2), the evaluator shall carry out the following steps: The evaluator shall permit a successful TLS handshake to occur in which a session ticket is exchanged with the non-TOE client. The evaluator shall then attempt to correctly reuse the previous session by sending the session ticket in the ClientHello. The evaluator shall confirm that the TOE responds with an abbreviated handshake described in Section 3.1 of RFC 5077 and illustrated with an example in figure 2. Of particular note: if the server successfully verifies the client's ticket, then it may renew the ticket by including a NewSessionTicket

	handshake message after the ServerHello in the abbreviated handshake (which is shown in figure 2). This is not required, however as further clarified in Section 3.3 of RFC 5077. When the session is resumed, the evaluator shall verify on the TLS Client used for performing this test, that the TOE (TLS Server) has not advertised support for the early data extension.
Pass/Fail with Explanation	N/A. This test is not applicable since TOE does not support session resumption using session tickets according to RFC 5077 (TLS 1.2).

7.1.2.85 FCS_TLSS_EXT.1.4 TEST #3B (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	*Test Objective: To demonstrate that the TOE will not resume a session for which the client failed to complete the handshake (independent of TOE support for session resumption).* Test 3 [conditional]: If the TOE supports session tickets according to RFC 5077 (supported only by TLS 1.2), the evaluator shall carry out the following steps: i. The evaluator shall permit a successful TLS handshake to occur in which a session ticket is exchanged with the non-TOE client. The evaluator shall then modify the session ticket and send it as part of a new Client Hello message. The evaluator shall confirm that the TOE either (1) implicitly rejects the session ticket by performing a full handshake (as shown in figure 3 or 4 of RFC 5077), or (2) terminates the connection in some way that prevents the flow of application data. Remark: If multiple contexts are supported for session resumption, for each of the above test cases, the session ticket may be obtained in one context for resumption in another context. There is no requirement that the session ticket be obtained and replayed within the same context subject to the description provided in the TSS. All contexts that can reuse a session ticket constructed in another context must be tested. It is not mandated that the session establishment and session resumption share context. For example, it is acceptable for a control channel to establish and application channel to resume the session.
Pass/Fail with Explanation	N/A. This test is not applicable since TOE does not support session resumption using session tickets according to RFC 5077 (TLS 1.2).

7.1.2.86 FCS_TLSS_EXT.1.4 TEST #4A (CPP_ND_V3.0E)

Item	Data

Test Assurance Activity	*Test Objective: To demonstrate that the TOE will not resume a session for which the client failed to complete the handshake (independent of TOE support for session resumption).* Test 4 [conditional]: If the TOE supports session resumption according to RFC 8446 (supported only by TLS 1.3), the evaluator shall carry out the following steps: i. The evaluator shall permit a successful TLS handshake to occur in which a session ticket is exchanged with the non-TOE client. The evaluator shall then attempt to correctly reuse the previous session by sending the pre-shared key in the ClientHello. The evaluator shall confirm that the TOE responds similarly to figure 3 of RFC 8446 after successfully reusing the pre-shared-key to resume the session. Specifically, the server must not send back a Certificate message if the session is correctly resumed. When the session is resumed, the evaluator shall verify on the TLS Client used for performing this test, that the TOE (TLS Server) has not advertised support for the early data extension.
Pass/Fail with Explanation	N/A. This test is not applicable since TOE does not support session resumption according to RFC 8446 and does not support TLS 1.3.

7.1.2.87 FCS_TLSS_EXT.1.4 TEST #4B (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	*Test Objective: To demonstrate that the TOE will not resume a session for which the client failed to complete the handshake (independent of TOE support for session resumption).* Test 4 [conditional]: If the TOE supports session resumption according to RFC 8446 (supported only by TLS 1.3), the evaluator shall carry out the following steps: ii. The evaluator shall permit a successful TLS handshake to occur in which a session ticket is exchanged with the non-TOE client. The evaluator shall then modify the pre-shared key and send it as part of a new Client Hello message. The evaluator shall confirm that the TOE either (1) implicitly rejects the session ticket by performing a full handshake, or (2) terminates the connection in some way that prevents the flow of application data.
Pass/Fail with Explanation	N/A. This test is not applicable since TOE does not support session resumption according to RFC 8446 and does not support TLS 1.3.

7.1.2.88 FCS_TLSS_EXT.1.4 TEST #4C (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	*Test Objective: To demonstrate that the TOE will not resume a session for which the client failed to complete the handshake (independent of TOE support for session resumption).* Test 4 [conditional]: If the TOE supports session resumption according to RFC 8446 (supported only by TLS 1.3), the evaluator shall carry out the following steps: iii. The evaluator shall permit a successful TLS handshake to occur in which a session ticket is exchanged with the non-TOE client. The evaluator shall then force the non-TOE client to attempt to establish a new connection using the previous session ticket material as a pre-shared key, but set psk_key_exchange_modes with a value of psk_ke in the Client Hello message and omit the psk_ke_dhe. The evaluator shall confirm that the TOE either (1) implicitly rejects the session ticket by performing a full handshake, or (2) terminates the connection in some way that prevents the flow of application data.
Pass/Fail with Explanation	N/A. This test is not applicable since TOE does not support session resumption according to RFC 8446 and and does not support TLS 1.3.

7.1.2.89 FCS_TLSS_EXT.1.5 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	Test 1[conditional]: If the TSF provides the ability of configuring the list of supported ciphersuites, the evaluator shall establish a TLS connection using one of the possible configurations of the list of supported ciphersuites. The evaluator shall then change the configuration and repeat the test. The evaluator shall verify that the behavior of the TOE has changed according to the modification of the list of ciphers. This test shall be repeated for all supported TLS versions. If the TSF does not provide the ability of configuring the list of supported ciphersuites, this test shall be omitted.
Test Steps	**Part1:** • Configure the TOE with the list of supported ciphersuites and try to establish a connection using TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA • Establish a connection with the TOE over TLS using the configured ciphersuite and verify a successful connection. • Verify through packet capture that the connection was established using the configured ciphersuite. **Part2:** • Remove the ciphersuite TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA from the list of supported ciphersuites. • Verify the cipher is removed via TOE logs.

	• Establish a connection with the TOE over TLS using ciphersuite TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA and verify that the connection fails. • Verify connection failure through the TOE logs. • Verify connection failure through the packet capture.
Pass/Fail with Explanation	Pass. The behavior of the TOE changes according to the modification of the list of ciphers. This test is performed for all supported TLS versions. This meets the testing requirements.

7.1.2.90 FCS_TLSS_EXT.1.6 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	According to RFC 8446 Section 4.2.10, a PSK is required to use the early data extension. As NDcPP only allows the use of PSK in conjunction with session resumption, a NDcPP conformant TOE which acts as TLS Server cannot use the early data extension if session resumption is not supported. For TOEs that do not support session resumption, execution of test FCS_TLSS_EXT.1.4 Test 1 is regarded as sufficient that the TOE does not support the early data extension. For TOEs that support session resumption, FCS_TLSS_EXT.1.4 Test 2(i), 3(i) or 4(i) (depending on the supported TLS versions and the way session resumption is implemented) ensure that the TOE does not support the early data extension.
Pass/Fail with Explanation	Pass. The test is performed under FCS_TLSS_EXT.1.4 Test #1, demonstrating that the TOE does not support session resumption for TLS 1.2.

7.1.2.91 FCS_TLSS_EXT.1.7 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	No test assurance activities have been identified.
Pass/Fail with Explanation	Pass. There are no test assurance activities for this SFR.

7.1.2.92 FCS_TLSS_EXT.1.8 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests:

	Test 1 [conditional]: If "support secure renegotiation…" is selected, the evaluator shall use a network packet analyzer/sniffer to capture a TLS 1.2 handshake between the two TLS endpoints. The evaluator shall verify that the "renegotiation_info" extension is included in the ServerHello message.
Pass/Fail with Explanation	N/A. The TOE does not "support secure renegotiation".

7.1.2.93 FCS_TLSS_EXT.1.8 TEST #2 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 2 [conditional]: If "support secure renegotiation…" is selected, the evaluator shall perform a TLS 1.2 handshake and modify the length portion of the field in the ClientHello message in the initial handshake to be non-zero. The evaluator shall verify that the TOE TLS server sends a failure and terminates the connection. The evaluator shall verify that a properly formatted field results in a successful TLS connection.
Pass/Fail with Explanation	N/A. The TOE does not "support secure renegotiation".

7.1.2.94 FCS_TLSS_EXT.1.8 TEST #3 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 3 [conditional]: If "support secure renegotiation…" is selected, the evaluator shall perform a TLS 1.2 handshake and modify the "client_verify_data" or "server_verify_data" value in the ClientHello message received during secure renegotiation. The evaluator shall verify that the TOE TLS server terminates the connection.
Pass/Fail with Explanation	N/A. The TOE does not "support secure renegotiation".

7.1.2.95 FCS_TLSS_EXT.1.8 TEST #4A **[TD0899]** (CPP_ND_V3.0E)

Item	Data

Test Assurance Activity	The evaluator shall perform the following tests: Test 4a [conditional, if the TOE supports TLS 1.3]: The evaluator shall follow the operational guidance as necessary to configure the TSF to negotiate TLS 1.3. The evaluator shall initiate a valid initial TLS 1.3 session, send a valid client hello on the non-renegotiable TLS channel, and observe that the TSF terminates the session. Note: It is preferred that the TSF sends a fatal error alert message (e.g., unexpected message) in response to this, but it is acceptable that the TSF terminates the connection silently (i.e., without sending a fatal error alert). **TD0899 has been applied.**
Pass/Fail with Explanation	N/A. Support for TLS version 1.3 is not claimed by ST.

7.1.2.96 FCS_TLSS_EXT.1.8 TEST #4B **[TD0899]** (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 4b [conditional, if the TOE supports TLS 1.2 and rejects TLS 1.2 renegotiation attempts]: The evaluator shall follow the operational guidance as necessary to configure the TSF to negotiate TLS 1.2 and reject renegotiation. The evaluator shall initiate a valid initial TLS 1.2 session, send a valid ClientHello on the non-renegotiable TLS channel, and observe that the TSF does not perform renegotiation of the TLS channel. **TD0899 has been applied.**
Test Steps	• Using the 'acumen-tls' tool send a valid ClientHello message to start a session using TLS version 1.2 and monitor the connection to ensure the TOE terminates the session after receiving the renegotiation request. • Verify through packet capture that the TOE terminates the session after receiving the renegotiation request. • Verify through TOE logs that TOE terminates the session after receiving the renegotiation request.
Pass/Fail with Explanation	Pass. The TOE correctly negotiates the specified TLS version and terminates the session when renegotiation is attempted. This meets the testing requirements.

7.1.3 IDENTIFICATION AND AUTHENTICATION (FIA)

7.1.3.1 FIA_AFL.1 TEST #1 (CPP_ND_V3.0E)

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 reenabled). The evaluator shall test that once the authentication attempts limit is reached, authentication attempts with valid credentials are no longer successful.
Test Steps	WUI: • Configure the TOE to lockout a user after 3 attempts. • Confirm the configuration has been implemented in the config. • Attempt to connect to the TOE with incorrect credentials until the limit is reached. • Verify that attempts with unsuccessful credentials will be rejected. • Attempt to connect to the TOE with correct credentials. • Verify that attempts with successful credentials will be rejected after the authentication failure limit is reached. WUI with LDAP Authentication: • Configure the TOE to lockout a user after 3 attempts. • Confirm the configuration has been implemented in the config. • Attempt to connect to the TOE with incorrect credentials until the limit is reached. • Verify that attempts with unsuccessful credentials will be rejected. • Attempt to connect to the TOE with correct credentials. • Verify that attempts with successful credentials will be rejected after the authentication failure limit is reached.
Pass/Fail with Explanation	Pass. An Administrator is not able to log into the TOE once the maximum authentication failure limit is reached even after using correct password. This meets the testing requirements.

7.1.3.2 FIA_AFL.1 TEST #2A (CPP_ND_V3.0E)

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).
Test Steps	WUI User: • Login into the TOE using administrator account. • Verify that the account from FIA_AFL.1 TEST #1 is locked out. • Verify that the account is locked out via Log. • Manually unlock the account using the administrator account. • Verify the account was unblocked via logs. • Authenticate using previously locked account and verify correct credentials result in access to the TOE. • Verify access to the TOE was established via logs. LDAP User: • Login into the TOE using administrator account. • Verify that the account from FIA_AFL.1 TEST #1 is locked out. • Verify that the account is locked out via Log. • Manually unlock the account using the administrator account. • Verify the account was unblocked via logs. • Authenticate using previously locked account and verify correct credentials result in access to the TOE. • Verify access to the TOE was established via logs.
Pass/Fail with Explanation	Pass. The TOE allows an administrator to manually re-enable the remote administrator's access (via WEBUI) which was locked out due to authentication failure. This meets the testing requirements.

7.1.3.3 FIA_AFL.1 TEST #2B (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests for each method by which remote administrators access the TOE (e.g. any passwords entered as part of establishing the connection protocol or the remote administrator application):

	Test 2: After reaching the limit for unsuccessful authentication attempts as in Test 1 above, the evaluator shall proceed as follows. If the **time period** selection in FIA_AFL.1.2 is included in the ST, then the evaluator shall wait for just less than the time period configured in Test 1 and show that an authorisation attempt using valid credentials does not result in successful access. The evaluator shall then wait until just after the time period configured in Test 1 and show that an authorisation attempt using valid credentials results in successful access
Pass/Fail with Explanation	N/A. Time Period Selection is not claimed in ST.

7.1.3.4 FIA_PMG_EXT.1 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests. Test 1: The evaluator shall compose passwords that meet the requirements in some way. For each password, the evaluator shall verify that the TOE supports the password. While the evaluator is not required (nor is it feasible) to test all possible compositions of passwords, the evaluator shall ensure that all characters, and a minimum length listed in the requirement are supported and justify the subset of those characters chosen for testing
Test Steps	• Set the minimum password length to 8 characters. • Verify via logs that the configuration is updated. • Create a user “good1” with a combination of lowercase, uppercase letters, numbers and special characters --- {QweRty!@#$%^789} • Verify that the new user has been created. • Verify via logs that the user has been created. • Create a user “good2” with a combination of lowercase, uppercase letters, numbers and special characters--- “(‘Sun,456-./’)&*+;” • Verify that the new user has been created. • Verify via logs that the user has been created. • Create a user “good3” with a combination of lowercase, uppercase letters, numbers and special characters--- [“m0on”:<+?>123_`~|=\] • Verify that the new user has been created. • Verify via logs that the user has been created. • Create a user “good4” with a combination of lowercase, uppercase letters, numbers and special characters --- AbCdEfGhIjKlMnOpQrStUvWxYz1! • Verify that the new user has been created.

	• Verify via logs that the user has been created. • Create a user “good5” with a combination of lowercase, uppercase letters, numbers and special characters --- aBcDeFgHiJkLmNoPqRsTuVwXyZ2@ • Verify that the new user has been created. • Verify via logs that the user has been created.
Pass/Fail with Explanation	Pass. The TOE allows the creation of all combinations of uppercase, lowercase, special characters, and numerical characters as specified in the ST, thereby fulfilling the password policy requirements. This meets the testing requirements.

7.1.3.5 FIA_PMG_EXT.1 TEST #2 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests. Test 2: The evaluator shall compose passwords that do not meet the requirements in some way. For each password, the evaluator shall verify that the TOE does not support the password. While the evaluator is not required (nor is it feasible) to test all possible compositions of passwords, the evaluator shall ensure that the TOE enforces the allowed characters and the minimum length listed in the requirement and justify the subset of those characters chosen for testing.
Test Steps	• Set the minimum password length to 8 characters. • Verify via logs that the configuration is updated. • Create a user “bad1” with 7 characters using the combination of lowercase and uppercase letters and numbers “12390Rt”. • Verify that the user is not created. • Create a user “bad2” with 6 characters using numbers and special characters “12%$34”. • Verify that the user is not created. • Create a user “bad3” with 4 characters using the combination of lowercase, uppercase letters, numbers and special characters “A#1k”. • Verify that the new user is not created.
Pass/Fail with Explanation	Pass. The TOE does not allow passwords that do not meet the requirements to be set for users. This meets the testing requirements.

7.1.3.6 FIA_UIA_EXT.1 TEST #1 (CPP_ND_V3.0E)

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 all combinations of supported credentials and login methods, 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 and verify that it fails. • Verify that an audit record was generated showing login failure. • Log into the TOE from a local connection with correct credentials and verify that it is successful. • Verify that an audit record was generated showing login success. WUI: • Attempt to login to TOE remote WUI connection with incorrect credentials and verify that it fails. • Verify that an audit record was generated showing login failure. • Log into the TOE remote WUI connection with correct credentials and verify that it is successful. • Verify that an audit record was generated showing login success. WUI with LDAP Authentication: • Attempt to login to TOE remote WUI connection with incorrect credentials and verify that it fails. • Verify that an audit record was generated showing login failure. • Log into the TOE remote WUI connection with correct credentials and verify that it is successful. • Verify that an audit record was generated showing login success.
Pass/Fail with Explanation	Pass. The TOE does not allow login with incorrect credentials while allowing login with correct credentials for each method [Local and Remote (WUI, External Authentication Server)] by which administrators access the TOE. This meets the testing requirements.

7.1.3.7 FIA_UIA_EXT.1 TEST #2 (CPP_ND_V3.0E)

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	**1.** Display the warning banner in accordance with FTA_TAB.1: WUI: • Verify that no functionality except those specified in the requirement is allowed. • Verify that after clicking 'Accept' icon, the page will be displayed asking for login credentials and no other system services can be accessed. • Login into the TOE via WUI. • Verify that the services are available after the login. • Verify authentication logs reflect success. WUI with LDAP Authentication: • Verify that no functionality except those specified in the requirement is allowed. • Verify that after clicking 'Accept' icon, the page will be displayed asking for login credentials and no other system services can be accessed. • Login into the TOE via WUI. • Verify that the services are available after the login. • Verify authentication logs reflect success. **2.** Automated generation of cryptographic keys: This requirement is fulfilled in accordance with test FPT_TUD_EXT.1 Test#1. In the logs, we can see that the TOE generated a self-signed certificate key after a successful reboot after updating the software.
Pass/Fail with Explanation	Pass. Successful identification and authentication on the TOE by an external remote entity enables the services on the TOE. This meets the testing requirements.

7.1.3.8 FIA_UIA_EXT.1 TEST #3 (CPP_ND_V3.0E)

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	**1.** Display the warning banner in accordance with FTA_TAB.1: • Configure a notice and consent warning message on the TOE. • Verify that the audit logs reflect the configuration of the access banner. • Verify that no functionality except those specified in the requirement is allowed. • Verify that when connected via console, the TOE displays the login banner and asks for login credentials and no other system services can be accessed. • Login into the TOE via console. • Verify that the services are available after the login. • Verify authentication logs reflect success. **2.** Automated generation of cryptographic keys: This requirement is fulfilled in accordance with test FPT_TUD_EXT.1 Test#1. In the logs, we can see that the TOE generated a self-signed certificate key after a successful reboot after updating the software.
Pass/Fail with Explanation	Pass. Services available to local access on the TOE should be enabled only after successful authentication and identification. This meets the testing requirements.

7.1.3.9 FIA_UIA_EXT.1 TEST #4 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests for each method by which administrators access the TOE (local and remote), as well as for each type of credential supported by the login method: Test 4: For distributed TOEs where not all TOE components support the authentication of Security Administrators according to FIA_UIA_EXT.1, the evaluator shall test that the components authenticate Security Administrators as described in the TSS.
Pass/Fail with Explanation	N/A. The TOE is not distributed; hence, this activity is not applicable.

7.1.3.10 FIA_UAU.7 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following test for each method of local login allowed: Test 1: 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	• Login into TOE with incorrect password and verify that obscured feedback is provided. • Verify login attempt via logs. • Login into TOE with correct password and verify that obscured feedback is provided. • Verify login attempt via logs.
Pass/Fail with Explanation	Pass. The TOE provides obscured feedback while entering the authentication information. This meets the testing requirements.

7.1.3.11 FIA_X509_EXT.1.1/REV TEST #1A (CPP_ND_V3.0E)

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 expected that either OCSP or CRL revocation checking is performed when a certificate is presented to the TOE (e.g. during authentication). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols: 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 valid 3-length chain of certificates using the XCA tool. • Import the CA and ICA to the TOE's trust store.

	• Verify the certificates were imported via TOE logs. • Utilize acumen-tls tool to start a TLS server using valid certificate. • Initiate a TLS connection between TOE and TLS server. • Verify the successful TLS connection logs via TOE logs. • Verify the successful TLS connection via packet capture.
Pass/Fail with Explanation	Pass. TOE accepts a TLS connection with a valid chain of x509 certificates. This meets the testing requirements.

7.1.3.12 FIA_X509_EXT.1.1/REV TEST #1B (CPP_ND_V3.0E)

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 expected that either OCSP or CRL revocation checking is performed when a certificate is presented to the TOE (e.g. during authentication). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols:i Test 1b: The evaluator shall then 'break' the chain used in Test 1a by either removing the trust anchor in the TOE's trust store used to terminate the chain, or by removing one of the intermediate CA certificates (provided together with the leaf certificate in Test 1a) to complete the chain. The evaluator shall show that an attempt to validate this broken chain fails.
Test Steps	• Remove the Root CA certificate from the TOE's trust store. • Verify via TOE logs the CA certificate was removed. • Utilize acumen-tls tool to start a TLS server using valid certificate. • Initiate a TLS connection between TOE and TLS server. • Verify the unsuccessful TLS connection via TOE logs. • Verify the unsuccessful TLS connection via packet capture.
Pass/Fail with Explanation	Pass. The TOE rejects a TLS connection with an invalid certificate path. This meets the testing requirements.

7.1.3.13 FIA_X509_EXT.1.1/REV TEST #2 (CPP_ND_V3.0E)

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 expected that either OCSP or CRL revocation checking is performed when a certificate is presented to the TOE (e.g. during authentication). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols:i Test 2: The evaluator shall demonstrate that validating an expired certificate results in the function failing.
Test Steps	• Create an expired server certificate using the XCA tool. • Show current time on the TOE. • Utilize acumen-tls tool to start a TLS server using expired certificate. • Initiate a TLS connection between TOE and TLS server. • Verify via TOE logs the connection was unsuccessful. • Verify via packet capture the connection was unsuccessful.
Pass/Fail with Explanation	Pass. TOE reject a TLS connection for an expired server x509 certificate. This meets the testing requirements.

7.1.3.14 FIA_X509_EXT.1.1/REV TEST #3 (CPP_ND_V3.0E)

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 expected that either OCSP or CRL revocation checking is performed when a certificate is presented to the TOE (e.g. during authentication). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols:i 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 shall then attempt 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	Note: CRL is not selected in ST. 1. Valid Certificate: • Create certificates with OCSP EKU and configure URI of the OCSP responder. • Load the root CA on the TOE. • Configure the TOE for OCSP checking. • Configure the TOE for syslog server. • Start OCSP responder for ICA and Server certificates. • Initiate the connection between Syslog server and the TOE using Server and ICA certificates. • Verify the successful connection via logs on the TOE. • Verify the successful connection with packet capture. 2. Invalid End Entity Certificate: • Revoke the End Entity certificate. • Start OCSP responder for ICA and Server certificates. • Initiate the connection between Syslog server and the TOE using Server and ICA certificates. • Verify the unsuccessful connection via logs on the TOE. • Verify the unsuccessful connection with packet capture. 3. Invalid Intermediate CA Certificate: • Reset the certificate chain and revoke only the intermediate CA certificate. • Start OCSP responder for ICA and Server certificates. • Initiate the connection between Syslog server and the TOE using Server and ICA certificates. • Verify the unsuccessful connection via logs on the TOE. • Verify the unsuccessful connection with packet capture.
Pass/Fail with Explanation	Pass. TOE rejects the TLS connection for the revoked certificate and accepts the TLS connection for a valid certificate path. This meets the testing requirements.

7.1.3.15 FIA_X509_EXT.1.1/REV TEST #4A (CPP_ND_V3.0E)

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 expected that either OCSP or CRL revocation checking is performed when a certificate is presented to the TOE (e.g. during authentication). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for

	each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols: Test 4a: [conditional] If OCSP is selected, the evaluator shall configure an authorized responder or use a man-in-the-middle tool to use a delegated OCSP signing authority to respond to the TOE's OCSP request. The resulting positive OCSP response (certStatus: good (0)) shall be signed by an otherwise valid and trusted certificate with the extendedKeyUsage extension that does not contain the OCSPSigning (OID 1.3.6.1.5.5.7.3.9). The evaluator shall verify that the TSF does not successfully complete the revocation check. Note: Per RFC 6960 Section 4.2.2.2, the OCSP signature authority is delegated when the CA who issued the certificate in question is NOT used to sign OCSP responses.
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 unsuccessful connection via logs on the device. • Verify the unsuccessful connection via packet capture.
Pass/Fail with Explanation	Pass. The TOE does not connect to the TLS server if OCSP signing purpose is missing in OCSP responder certificate.

7.1.3.16 FIA_X509_EXT.1.1/REV TEST #4B (CPP_ND_V3.0E)

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 expected that either OCSP or CRL revocation checking is performed when a certificate is presented to the TOE (e.g. during authentication). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols:i Test 4b: [conditional] If CRL is selected, the evaluator shall present an otherwise valid CRL signed by a trusted certificate that does not have the cRLsign key usage bit set and verify that validation of the CRL fails.
Pass/Fail with Explanation	N/A. CRL is not selected in the ST.

7.1.3.17 FIA_X509_EXT.1.1/REV TEST #5 (CPP_ND_V3.0E)

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 expected that either OCSP or CRL revocation checking is performed when a certificate is presented to the TOE (e.g. during authentication). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols:i Test 5: 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	• Utilize Acumen-tls tool to modify any byte in the first eight bytes of the server certificate. • Verify the unsuccessful TLS connection via TOE logs. • Verify the unsuccessful TLS connection via packet capture.
Pass/Fail with Explanation	Pass. TOE rejects the TLS connection when the first 8 bytes of the certificate are modified. This meets the testing requirements.

7.1.3.18 FIA_X509_EXT.1.1/REV TEST #6 (CPP_ND_V3.0E)

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 expected that either OCSP or CRL revocation checking is performed when a certificate is presented to the TOE (e.g. during authentication). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols:i Test 6: The evaluator shall modify any byte in the certificate signatureValue field (see RFC 5280 Section 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	• Utilize Acumen-tls tool to modify any byte in the certificate signatureValue field. • Verify the unsuccessful TLS connection via TOE logs. • Verify the unsuccessful TLS connection via packet capture.

Pass/Fail with Explanation	Pass. TOE rejects TLS connection for a certificate due to modification in the certificate signatureValue field. This meets the testing requirements.

7.1.3.19 FIA_X509_EXT.1.1/REV TEST #7 (CPP_ND_V3.0E)

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 expected that either OCSP or CRL revocation checking is performed when a certificate is presented to the TOE (e.g. during authentication). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols:i 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	• Utilize Acumen-tls tool to modify any byte in the public key of the certificate. • Start a TLS connection from the toe to the Acumen-TLS tool. • Verify the unsuccessful TLS connection via TOE logs. • Verify the unsuccessful TLS connection via packet capture.
Pass/Fail with Explanation	Pass. TOE rejects a TLS connection for a certificate due to modification in the public key of the certificate. This meets the testing requirements.

7.1.3.20 FIA_X509_EXT.1.1/REV TEST #8A (CPP_ND_V3.0E)

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 expected that either OCSP or CRL revocation checking is performed when a certificate is presented to the TOE (e.g. during authentication). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols:

	Test 8 (Conditional on support for EC certificates as indicated in FCS_COP.1/SigGen). The following tests are run when a minimum certificate path length of three certificates is implemented: Test 8a: (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.
Pass/Fail with Explanation	Pass. The requirement for this test is covered in FIA_X509_EXT.1.1/Rev Test #1. This meets the testing requirements.

7.1.3.21 FIA_X509_EXT.1.1/REV TEST #8B (CPP_ND_V3.0E)

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 expected that either OCSP or CRL revocation checking is performed when a certificate is presented to the TOE (e.g. during authentication). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols:i Test 8 (Conditional on support for EC certificates as indicated in FCS_COP.1/SigGen). The following tests are run when a minimum certificate path length of three certificates is implemented: Test 8b: (Conditional on TOE ability to process CA certificates presented in certificate message) The test shall be designed in a way such that only the EC root certificate is designated as a trust anchor, and by setting up the trust store in a way that the EC Intermediate CA certificate needs to be provided, together with the leaf certificate, from outside the TOE to complete the chain (e.g. by storing only the EC root CA certificate in the trust store). The evaluator shall present the TOE with a chain of EC certificates (terminating in a trusted CA certificate), where the intermediate certificate in the certificate chain uses an explicit format version of the Elliptic Curve parameters in the public key

	information field, and is signed by the trusted EC root CA, but having no other changes. The evaluator shall confirm the TOE treats the certificate as invalid.
Test Steps	• In the Second part of the test Intermediate certificate is modified with a named curve with an explicit format in the public key information field. • Concatenate the CA certificates. • Add server certificate to the modified chain. • Configure the TOE for the root certificate as a trust anchor. • Attempt the connection from the TOE to the TLS Server. • Verify the failure logs on the device. • Verify the unsuccessful connection via packet capture.
Pass/Fail with Explanation	Pass. The evaluator verified that when the public key information is modified with a named curve with an explicit format in the intermediate certificate on the TLS server, TOE is unable to make the successful connection. This meets the test requirements.

7.1.3.22 FIA_X509_EXT.1.1/REV TEST #8C (CPP_ND_V3.0E)

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 expected that either OCSP or CRL revocation checking is performed when a certificate is presented to the TOE (e.g. during authentication). The evaluator shall perform the following tests for FIA_X509_EXT.1/Rev. These tests must be repeated for each distinct security function that utilizes X.509v3 certificates. For example, if the TOE implements certificate-based authentication with IPSEC and TLS, then it shall be tested with each of these protocols:i Test 8 (Conditional on support for EC certificates as indicated in FCS_COP.1/SigGen). The following tests are run when a minimum certificate path length of three certificates is implemented: Test 8c: The evaluator shall establish a subordinate CA certificate, where the elliptic curve parameters are specified as a named curve, that is signed by a trusted EC root CA. The evaluator shall attempt to load the certificate into the trust store and observe that it is accepted into the TOE's trust store. The evaluator shall then establish a subordinate CA certificate that uses an explicit format version of the elliptic curve parameters, and that is signed by a trusted EC root CA. The evaluator shall attempt to load the certificate into the trust store and observe that it is rejected, and not added to the TOE's trust store.
Test Steps	• In the Third part of the test Intermediate certificate is modified with a named curve with an explicit format in the public key information field. • Attempt to add the modified Intermediate certificate on the TOE.

	• Verify that the TOE discards the certificate. • Verify that the log that the TOE discards the certificate.
Pass/Fail with Explanation	Pass. The evaluator verified that when the public key information is modified with a named curve with an explicit format in the intermediate certificate and is loaded to the TOE's trust store, TOE does not accept such certificate. This meet the testing requirements.

7.1.3.23 FIA_X509_EXT.1.2/REV TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests for FIA_X509_EXT.1.2/Rev. 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 shall confirm that the TOE rejects such a certificate at one (or both) of the following points: (i) as part of the validation of the leaf certificate belonging to this chain; (ii) when attempting to add a CA certificate without the basicConstraints extension to the TOE's trust store (i.e. when attempting to install the CA certificate as one which will be retrieved from the TOE itself when validating future certificate chains). The evaluator shall repeat these tests for each distinct use of certificates. Thus, for example, use of certificates for TLS connection was distinct from use of certificates for trusted updates so both of these uses would be tested. But there is no need to

	repeat the tests for each separate TLS channel in FTP_ITC.1 and FTP_TRP.1/Admin (unless the channels use separate implementations of TLS).
Test Steps	• Configure the ICA certificate lacking the basicConstraints extension. • Load the certificate lacking the basicConstraints on the TLS server. • Add the modified certificate to the certificate chain. • Concatenate the CA certificates. • Show the concatenated certificate. • Utilize Acumen-tls tool to make a TLS connection with the TOE. • Verify the error in logs on the device. • Verify the unsuccessful connection with packet capture.
Pass/Fail with Explanation	Pass. TOE rejects CA certificates lacking the basicConstraints extension. This meets the testing requirements.

7.1.3.24 FIA_X509_EXT.1.2/REV TEST #2 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests for FIA_X509_EXT.1.2/Rev. 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 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 shall confirm that the TOE rejects such a certificate at one (or both) of the following points:

	(i) as part of the validation of the leaf certificate belonging to this chain; (ii) when attempting to add a CA certificate with the CA flag set to FALSE to the TOE's trust store (i.e. when attempting to install the CA certificate as one which will be retrieved from the TOE itself when validating future certificate chains). The evaluator shall repeat these tests for each distinct use of certificates. Thus, for example, use of certificates for TLS connection is distinct from use of certificates for trusted updates so both of these uses would be tested. But there is no need to repeat the tests for each separate TLS channel in FTP_ITC.1 and FTP_TRP.1/Admin (unless the channels use separate implementations of TLS).
Test Steps	• Configure the CA certificate with the flag in the basicConstraints extension set to FALSE. • Add the modified certificate to the certificate chain. • Concatenate the CA certificates. • Attempt the connection from the TOE to the TLS Server. • Verify the error in logs on the device. • Verify the unsuccessful connection packet capture.
Pass/Fail with Explanation	Pass. The TOE rejects certificates signed by a CA that has the CA flag in the basicConstraints extension set to FALSE. This meets the test requirements.

7.1.3.25 FIA_X509_EXT.2 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following test for each trusted channel: Test 1: The evaluator shall demonstrate that using a valid certificate that requires certificate validation checking to be performed in at least some part by communicating with a non-TOE IT entity. The evaluator shall then manipulate the environment so that the TOE is unable to verify the validity of the certificate and observe that the action selected in FIA_X509_EXT.2.2 is performed. If the selected action is administrator-configurable, then the evaluator shall follow the guidance documentation to determine that all supported administrator-configurable options behave in their documented manner
Test Steps	1. Valid Certificate: • Testing for valid certificate was completed in FIA_X509_EXT.1.1/Rev Test #3. 2. TOE is unable to validate the certificate from the OCSP server:

	In the second part of the test, the server certificate is revoked and the OCSP server is unreachable. The TOE does not make a successful connection with the TLS server if unable to validate the certificate. • Configure the server certificate which has been revoked. • 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. 3. With administrator-configurable option: In the third part of the test, the server certificate is revoked and the OCSP server is unreachable, but the TOE is configured to allow access on OCSP server failure. The TOE makes a successful connection with the TLS server even if the certificate validation fails. • Configure the server certificate which has been revoked. • 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. • Configure the option “Allow Access on Server Failure” on TOE. • Attempt the connection from the TOE to the TLS server and show the connection being successful. • Verify the logs on TOE. • Verify the packet capture between the TOE and the TLS server.
Pass/Fail with Explanation	Pass. The TOE rejects a TLS connection if unable to validate the server certificate. If administrator-configurable action is selected, the Administrator chooses whether to accept the certificate in these cases. This meets testing requirements.

7.1.3.26 FIA_X509_EXT.3 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 1: The evaluator shall use the guidance documentation to cause the TOE to generate a Certification Request. The evaluator shall capture the generated message and ensure that it conforms to the format specified. The evaluator shall

	confirm that the Certification Request provides the public key and other required information, including any necessary user-input information.

Test Steps	• On the TOE, generate a CSR.
	• Verify via TOE logs the CSR was generated.
	• Examine the CSR contents. Ensure the CSR contains the following:
	  ○ Public key
	  ○ Device specific information (e.g. email address, requested SAN name)
	  ○ Common Name
	  ○ Organization
	  ○ Organizational Unit
	  ○ Country

Pass/Fail with Explanation	Pass. TOE is able to generate CSR requests containing fields claimed in the ST. This meets the testing requirements.

7.1.3.27 FIA_X509_EXT.3 TEST #2 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 2: The evaluator shall demonstrate that validating a response message to a Certification Request without a valid certification path results in the function failing. The evaluator shall then load a certificate or certificates as trusted CAs needed to validate the certificate response message and demonstrate that the function succeeds.
Test Steps	• Generate a CSR (Certificate Signing Request) on the TOE. • Generate a signed certificate based on the generated CSR from an external CA. • Ensure that the full trust chain for the signed CA is not present on the TOE. • Attempt to load the signed certificate on the TOE. • Verify from the logs that the certificate is installed. • Verify that the option to choose the certificate is disabled i.e., it is grayed out and cannot be selected as the full trust chain of the CA is not present. • Add the intermediate certificate to the TOE certificate store to ensure that the TOE has a full certificate path. • Verify from the logs that intermediate certificate is installed. • Verify that the option to choose the certificate is now enabled i.e., not grayed out and can be selected since the full trust chain of the CA is present on the TOE. • Verify that the certificate can be used via logs.

Pass/Fail with Explanation	Pass. The TOE does not install the CSR certificate when the full trust chain is not present on the TOE while when the full trust chain is present on the TOE it installs the CSR certificate. This meets the testing requirements.

7.1.4 SECURITY MANAGEMENT (FMT)

7.1.4.1 FMT_MOF.1/FUNCTIONS (1) TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	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 perform the following tests: Test 1: 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	• Access the TOE as a lower privilege user. • Attempt to modify security parameters related to external audit server. • Verify failure to modify security parameters related to external audit server.
Pass/Fail with Explanation	Pass. The TOE does not allow a lower privilege user to modify security parameters related to the transmission of audit data to an external audit server. This meets testing requirements.

7.1.4.2 FMT_MOF.1/FUNCTIONS (1) TEST #2 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	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 perform the following tests: Test 2: 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	• Access the TOE as a Security Administrator. • Attempt to modify all security parameters related to external audit server. • Verify successful modification via logs.
Pass/Fail with Explanation	Pass. The TOE allows an Administrator to modify security parameters related to the transmission of audit data to an external audit server. This meets testing requirements.

7.1.4.3 FMT_MOF.1/FUNCTIONS (2) TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	If **'handling of audit data**' is selected from the second selection together with **'modify the behaviour of**' in the first selection The evaluator shall perform the following tests: Test 1: The evaluator shall try to modify all security related parameters for configuration of the handling of audit data without prior authentication as Security Administrator (by authentication as a user with no administrator privileges or without user authentication at all). Attempts to modify parameters without prior authentication should fail. According to the implementation no other users than the Security Administrator might be defined and without any user authentication the user might not be able to get to the point where the attempt can be executed. In that case it shall be demonstrated that access control mechanisms prevent execution up to the step that can be reached without authentication as Security Administrator. The term 'handling of audit data' refers to the different options for selection and assignments in SFRs FAU_STG_EXT.1.4, FAU_STG_EXT.1.5 and FAU_STG_EXT.2.
Pass/Fail with Explanation	N/A. 'handling of audit data' not selected in ST.

7.1.4.4 FMT_MOF.1/FUNCTIONS (2) TEST #2 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	If **'handling of audit data**' is selected from the second selection together with **'modify the behaviour of**' in the first selection The evaluator shall perform the following tests: Test 2: The evaluator shall try to modify all security related parameters for configuration of the handling of audit data with prior authentication as Security Administrator. The effects of the modifications should be confirmed. The term 'handling of audit data' refers to the different options for selection and assignments in SFRs FAU_STG_EXT.1.4, FAU_STG_EXT.1.5 and FAU_STG_EXT.2. The evaluator does not necessarily have to test all possible values of the security related parameters for configuration of the handling of audit data but at least one allowed value per parameter.
Pass/Fail with Explanation	N/A. 'handling of audit data' not selected in ST.

7.1.4.5 FMT_MOF.1/FUNCTIONS (3) TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	If **'audit functionality when Local Audit Storage Space is full**' is selected from the second selection together with **'modify the behaviour of**' in the first selection The evaluator shall perform the following tests: Test 1: The evaluator shall try to modify the behaviour when Local Audit Storage Space is full without prior authentication as Security Administrator (by authentication as a user with no administrator privileges or without user authentication at all). This attempt should fail. According to the implementation no other users than the Security Administrator might be defined and without any user authentication the user might not be able to get to the point where the attempt can be executed. In that case it shall be demonstrated that access control mechanisms prevent execution up to the step that can be reached without authentication as Security Administrator.

Pass/Fail with Explanation	N/A. 'audit functionality when Local Audit Storage Space is full' not selected in ST.

7.1.4.6 FMT_MOF.1/FUNCTIONS (3) TEST #2 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	If **'audit functionality when Local Audit Storage Space is full**' is selected from the second selection together with **'modify the behaviour of**' in the first selection The evaluator shall perform the following tests: Test 2: The evaluator shall try to modify the behaviour when Local Audit Storage Space is full with prior authentication as Security Administrator. This attempt should be successful. The effect of the change shall be verified. The evaluator does not necessarily have to test all possible values for the behaviour when Local Audit Storage Space is full but at least one change between allowed values for the behaviour.
Pass/Fail with Explanation	N/A. 'audit functionality when Local Audit Storage Space is full' not selected in ST.

7.1.4.7 FMT_MOF.1/FUNCTIONS (4) TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	If in the first selection **'determine the behaviour of**' has been chosen together with for any of the options in the second selection The evaluator shall perform the following tests: Test 1: The evaluator shall try to determine the behaviour of all options chosen from the second selection without prior authentication as Security Administrator (by authentication as a user with no administrator privileges or without user authentication at all). This can be done in one test or in separate tests. The attempt(s) to determine the behaviour of the selected functions without administrator authentication shall fail. According to the implementation no other users than the Security Administrator might be defined and without any user authentication the user might not be able to get to the point where the attempt can be executed. In that case it shall be demonstrated that access

	control mechanisms prevent execution up to the step that can be reached without authentication as Security Administrator.
Test Steps	• Access the TOE as a lower privilege user. • Attempt to determine security parameters related to external audit server. • Verify failure to determine security parameters related to external audit server.
Pass/Fail with Explanation	Pass. The TOE does not allow a lower privilege user to determine security parameters related to the transmission of audit data to an external audit server. This meets testing requirements.

7.1.4.8 FMT_MOF.1/FUNCTIONS (4) TEST #2 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	If in the first selection '**determine the behaviour of**' has been chosen together with for any of the options in the second selection The evaluator shall perform the following tests: Test 2: The evaluator shall try to determine the behaviour of all options chosen from the second selection with prior authentication as Security Administrator. This can be done in one test or in separate tests. The attempt(s) to determine the behaviour of the selected functions with Security Administrator authentication shall be successful.
Test Steps	• Access the TOE as a Security Administrator. • Attempt to determine security parameters related to external audit server. • Verify successful modification of security parameters of the external audit server of via logs.
Pass/Fail with Explanation	Pass. The TOE allows an Administrator to determine security parameters related to the transmission of audit data to an external audit server. This meets testing requirements.

7.1.4.9 FMT_MOF.1/MANUALUPDATE TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests:

	Test 1: 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	• Create an unprivileged user on the TOE. • Verify the user was created. • Verify the user was created via logs. • Login into the TOE with unprivileged user. • Verify that the user does not have the privilege to perform manual update on the TOE. • Login with privileged user on the TOE and show the option to perform manual update is available.
Pass/Fail with Explanation	Pass. The TOE does not allow the lower privilege users to update the TOE. This meets the testing requirements.

7.1.4.10 FMT_MOF.1/MANUALUPDATE TEST #2 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 2: The evaluator shall try to perform the update with prior authentication as Security Administrator using a legitimate update image. This attempt should be successful. This test case should be covered by the tests for FPT_TUD_EXT.1 already
Pass/Fail with Explanation	Pass. This test is performed in conjunction with FPT_TUD_EXT.1 where the Security Administrator successfully upgraded the TOE.

7.1.4.11 FMT_MOF.1/SERVICES TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 1: The evaluator shall try to enable and disable at least one of the services as defined in the Application Notes for FAU_GEN.1.1 (whichever is supported by the TOE) without prior authentication as Security Administrator (either by authenticating as a user with no administrator privileges, if possible, or without prior authentication at all). The attempt to enable/disable this service/these services 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 enable/disable this service/these services 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	Disable Syslog: • Access the TOE as a lower privilege user. • Attempt to disable logging to external syslog server. • Verify failure to disable syslog service. Disable WUI: • Access the TOE via console as a lower privilege user. Note: A lower privilege user is unable to access the TOE via console to disable the WUI service.
Pass/Fail with Explanation	Pass. The TOE does not allow a lower privilege user to disable system services. This meets testing requirements.

7.1.4.12 FMT_MOF.1/SERVICES TEST #2 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 2: The evaluator shall try to enable and disable at least one of the services as defined in the Application Notes for FAU_GEN.1.1 (whichever is supported by the TOE) with prior authentication as Security Administrator. The attempt to enable/disable this service/these services should be successful.
Test Steps	Disable Syslog: • Access the TOE as an Administrator. • Attempt to disable logging to external syslog server. • Verify success disabling service via TOE logs. • Attempt to enable logging to external syslog server. • Verify success enabling service via TOE logs. Disable WUI: • Access the TOE via console as an Administrator. • Attempt to disable WebUI service. • Verify success disabling service via TOE logs. • Attempt to enable WebUI service. • Verify success enabling service via TOE logs.
Pass/Fail with Explanation	Pass. The TOE allows an Administrator to enable and disable system services. This meets testing requirements.

7.1.4.13 FMT_MTD.1/COREDATA TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	No separate testing for FMT_MTD.1/CoreData is required unless one of the management functions has not already been exercised under any other SFR.
Pass/Fail with Explanation	Pass. No separate testing for FMT_MTD.1/CoreData is required as all management functions have already been already exercised under claimed SFRs and there are no remaining functions to be tested. This meets the testing requirements.

7.1.4.14 FMT_MTD.1/CRYPTOKEYS TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 1: 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	Crypto Key Generation using CSR: • Login into the TOE with unprivileged user. • Verify the generation of CSR fails for unprivileged user. Crypto Key Generation using Self-signed certificate: • Login into the TOE with unprivileged user. • Verify that the user does not have the privilege to generate crypto keys using self-signed certificate generation; this should fail. Crypto Key Generation via local Console: Note: An unprivileged user is unable to access the TOE via console.

	• Login into the TOE via console with unprivileged user. • Verify via logs the user is unable to access the TOE.
Pass/Fail with Explanation	Pass. Unprivileged user cannot generate the crypto keys using the Certificate Signing Requests and Self-signed certificate.

7.1.4.15 FMT_MTD.1/CRYPTOKEYS TEST #2 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 2: 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	Crypto Key Generation using CSR: • Login into the TOE with a privileged user. • Verify the generation of CSR passes for privileged user. • Verify via logs that the privileged user can generate CSR. Crypto Key Generation using Self-signed certificate: • Login into the TOE with privileged user. • Attempt to generate crypto keys using self-signed certificate generation; this will pass. • Verify the generation is successful via logs. Crypto Key Generation via local Console: • Login into the TOE via console with privileged user. • Attempt to generate cryptographic keys. • Verify the generation is successful via logs.
Pass/Fail with Explanation	Pass. Administrators can perform security related configurations on the TOE. This meets the testing requirements.

7.1.4.16 FMT_SMF.1 TEST #1 (CPP_ND_V3.0E)

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.

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 passed.

7.1.4.17 FMT_SMR.2 TEST #1 (CPP_ND_V3.0E)

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, if the TSF shall be validated against the Functional Package for Secure Shell referenced in Section 2.2 of the cPP; and TLS/HTTPS; then all three methods of administration must be exercised during the evaluation team's test activities.
Pass/Fail with Explanation	Pass, As there are two interfaces where these can be tested (over the GUI/Console) management functions are tested from the GUI interface. Each management function is not available on the Console interface. The evaluator has met this requirement through execution of the entirety of this test report for the TOE interfaces.

7.1.5 PROTECTION OF THE TSF (FPT)

7.1.5.1 FPT_APW_EXT.1 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	No test assurance activities have been identified.
Pass/Fail with Explanation	Pass. No test assurance activities have been identified.

7.1.5.2 FPT_SKP_EXT.1 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	No test assurance activities have been identified.

Pass/Fail with Explanation	Pass. No test assurance activities have been identified.

7.1.5.3 FPT_STM_EXT.1 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: 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. If the audit component of the TOE consists of several parts with independent time information, then the evaluator shall verify that the time information between the different parts are either synchronized or that it is possible for all audit information to relate the time information of the different part to one base information unambiguously.
Test Steps	• Login into the TOE with correct credentials via WUI. • Confirm the current time on the TOE. • Set a new time on the TOE via the remote WUI. • Verify that the new time is set. • Verify logs were generated for 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.

7.1.5.4 FPT_STM_EXT.1 TEST #2 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: 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 shall 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.

	If the audit component of the TOE consists of several parts with independent time information, then the evaluator shall verify that the time information between the different parts are either synchronized or that it is possible for all audit information to relate the time information of the different part to one base information unambiguously.
Test Steps	• Confirm the current time on the TOE. • Configure NTP support on the TOE via the remote console over GUI. • Verify that the new time is set. • Verify that the TOE time matches with the NTP server. • Verify via logs that the TOE synchronizes with the NTP server.
Pass/Fail with Explanation	Pass. The TOE was successfully able to synchronize with the NTP server. This meets the testing requirements.

7.1.5.5 FPT_STM_EXT.1 TEST #3 (CPP_ND_V3.0E)

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. If the audit component of the TOE consists of several parts with independent time information, then the evaluator shall verify that the time information between the different parts are either synchronized or that it is possible for all audit information to relate the time information of the different part to one base information unambiguously.
Pass/Fail with Explanation	N/A. The TOE does not obtain time from the underlying VS.

7.1.5.6 FPT_TST_EXT.1 TEST #1 (CPP_ND_V3.0E) **[TD0836]**

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], Section 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], Section 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 verify that the self-tests described above are carried out according to the SFR and in agreement with the descriptions in the TSS. 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. **TD0836 has been applied.**
Test Steps	Integrity and Cryptographic self-test during reboot: • Reboot the TOE and observe the TOE Start up. • Verify that the TOE performs self-test via TOE logs. Cryptographic self-test after enabling WUI: • Stop the web access service. • Start the web access service. • Verify that the TOE performs self-test via TOE logs.
Pass/Fail with Explanation	Pass. The TOE successfully executes self-test. This meets the testing requirement.

7.1.5.7 FPT_TUD_EXT.1 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 1: The evaluator shall perform 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 shall verify 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 shall perform the version verification activity again to verify the version correctly corresponds to that of the update and that current version of the product and most recently installed version match again. The evaluator shall perform Test 1 and Test 2 for all methods supported (manual updates, automatic checking for updates, automatic updates). For distributed TOEs the evaluator shall perform Test 1 and Test 2 for all TOE components.
Test Steps	• Verify the current version of the TOE. • Click on Update Software file and upload the image on the TOE. • Click on Verification file and upload the checksum file on the TOE. • Boot the uploaded image and wait for some time until the booting is done. • After reboot, show new version of software. • Verify the installation via logs.
Pass/Fail with Explanation	Pass. The TOE can be successfully updated. This meets the testing requirements.

7.1.5.8 FPT_TUD_EXT.1 TEST #2 (A) (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 2 [conditional]: If the TOE itself verifies a digital signature to authorize the installation of an image to update the TOE the following test shall be performed (otherwise the test shall be omitted). The evaluator shall first confirm that no updates are pending and then perform 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 shall obtain or produce illegitimate updates as defined below and attempt to install them on the TOE. The evaluator shall verify that the TOE rejects all of the illegitimate updates. The evaluator shall perform this test using all of the following forms of illegitimate updates:

	i. A modified version (e.g. using a hex editor) of a legitimately signed update 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 the current version and most recently installed version, reflect the same version information as prior to the update attempt.r The evaluator shall perform Test 1 and Test 2 for all methods supported (manual updates, automatic checking for updates, automatic updates). For distributed TOEs the evaluator shall perform Test 1 and Test 2 for all TOE components.
Test Steps	• Verify the current firmware version on the TOE. • Using a Hex editor modify an otherwise good firmware image. • Verify the current firmware version on the TOE. • Upload the modified image on the TOE. • Attempt to install the modified update image and verify that it fails. • Verify the failure with logs. • Verify that version has not changed.
Pass/Fail with Explanation	Pass. The TOE software was able to detect when an image was corrupted and rejected the image. This meets the testing requirements.

7.1.5.9 FPT_TUD_EXT.1 TEST #2 (B) (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 2 [conditional]: If the TOE itself verifies a digital signature to authorize the installation of an image to update the TOE the following test shall be performed (otherwise the test shall be omitted). The evaluator shall first confirm that no updates are pending and then perform 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 shall obtain or produce illegitimate updates as defined below and attempt to install them on the TOE. The evaluator shall verify that the TOE rejects all of the illegitimate updates. The evaluator shall perform this test using all of the following forms of illegitimate updates:

	ii. An image that has not been signed 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 the current version and most recently installed version, reflect the same version information as prior to the update attempt.r The evaluator shall perform Test 1 and Test 2 for all methods supported (manual updates, automatic checking for updates, automatic updates). For distributed TOEs the evaluator shall perform Test 1 and Test 2 for all TOE components.
Test Steps	• Verify the current version of the TOE. • Show blank verification signature file. • Upload the valid firmware image and blank digital signature file to the TOE. • Attempt to install an update with a blank verification signature file. • Verify the TOE rejects the software update. • Verify the error via logs. • Verify that version has not changed.
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.

7.1.5.10 FPT_TUD_EXT.1 TEST #2 (C) (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 2 [conditional]: If the TOE itself verifies a digital signature to authorize the installation of an image to update the TOE the following test shall be performed (otherwise the test shall be omitted). The evaluator shall first confirm that no updates are pending and then perform 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 shall obtain or produce illegitimate updates as defined below and attempt to install them on the TOE. The evaluator shall verify that the TOE rejects all of the illegitimate updates. The evaluator shall perform this test using all of the following forms of illegitimate updates:

	iii. 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) 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 the current version and most recently installed version, reflect the same version information as prior to the update attempt.r The evaluator shall perform Test 1 and Test 2 for all methods supported (manual updates, automatic checking for updates, automatic updates). For distributed TOEs the evaluator shall perform Test 1 and Test 2 for all TOE components.
Test Steps	• Verify the current version of the TOE. • Using a Hex editor modify to valid signature file. • Upload the valid firmware image and invalid digital signature file to the TOE. • Attempt to install an update with a modified digital signature file. • Verify the TOE rejects the software update due to invalid digital signature file. • Verify the error via logs. • Verify that version has not changed.
Pass/Fail with Explanation	Pass. The TOE software was able to detect when an image had an invalid signature and rejected the image. This meets the testing requirements.

7.1.6 TOE ACCESS (FTA)

7.1.6.1 FTA_SSL_EXT.1 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following test: Test 1: 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 shall establish a local interactive session with the TOE. The evaluator shall then observe that the session is either locked or terminated after the configured time period. If locking

	was selected from the component, the evaluator shall then ensure that reauthentication is needed when trying to unlock the session.
Test Steps	• Configure an inactivity time period of 5 minutes on TOE. • Verify log is generated for the configuration of the inactivity time period on TOE. • Login into TOE using local console. • Verify login time via log. • Wait for longer than 5 minutes and verify that the session expired due to the inactivity time period. • Verify via logs the session expired. • Configure an inactivity time period of 10 mins on TOE. • Verify log is generated for the configuration of the inactivity time period on TOE. • Login into TOE using local console. • Verify login time via log. • Wait for longer than 10 minutes and verify that the session expired due to the inactivity time period. • Verify via logs the session expired.
Pass/Fail with Explanation	Pass. TOE terminates a local interactive session after completion of the inactivity time period. This meets the testing requirements.

7.1.6.2 FTA_SSL.3 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	For each method of remote administration, the evaluator shall perform the following test: Test 1: 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 shall establish a remote interactive session with the TOE. The evaluator shall then observe that the session is terminated after the configured time period.
Test Steps	• Configure an inactivity time period of 5 minutes on TOE. • Verify log is generated for the configuration of the inactivity time period on TOE. • Login into TOE via WUI. • Verify login time via log. • Wait for longer than 5 minutes and verify that the session expired due to the inactivity time period. • Verify via logs the session expired.

	• Configure an inactivity time period of 10 mins on TOE. • Verify log is generated for the configuration of the inactivity time period on TOE. • Login into TOE via WUI. • Verify login time via log. • Wait for longer than 10 minutes and verify that the session expired due to the inactivity time period. • Verify via logs the session expired.
Pass/Fail with Explanation	Pass. TOE terminates a remote interactive session after completion of the inactivity time period. This meets the testing requirements.

7.1.6.3 FTA_SSL.4 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 1 [conditional]: If the TOE supports local administration, the evaluator shall initiate an interactive local session with the TOE. The evaluator shall then follow the guidance documentation to exit or log off the session and observes that the session has been terminated.
Test Steps	• Login into TOE using the local console. • Verify login via audit logs. • Logout of the local interactive session. • Verify logout via audit logs.
Pass/Fail with Explanation	Pass. The TOE allows the Administrator to terminate a local session using a session termination command. This meets the testing requirements.

7.1.6.4 FTA_SSL.4 TEST #2 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 2: For each method of remote administration, the evaluator shall initiate an interactive remote session with the TOE. The evaluator shall then follow the guidance documentation to exit or log off the session and observes that the session has been terminated.
Test Steps	• Login into TOE via WUI. • Verify login via audit logs. • Logout of the interactive remote session.

	• Verify logout via audit logs.
Pass/Fail with Explanation	Pass. The TOE allows the Administrator to terminate a remote session using a session termination command. This meets the testing requirements.

7.1.6.5 FTA_TAB.1 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall also perform the following test: Test 1: The evaluator shall follow 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	WUI: • Configure a notice and consent warning message on the TOE. • Verify that the audit logs reflect the configuration of the access banner. • Log onto the TOE remotely via WUI and ensure the access banner is present. Console: • Configure a notice and consent warning message on the TOE. • Verify that the audit logs reflect the configuration of the access banner. • Log onto the TOE via local access and ensure the access banner is present.
Pass/Fail with Explanation	Pass. The administrator is able to configure a notice and consent warning message for each login method (remote and local). This meets the testing requirements.

7.1.7 TRUSTED PATH/CHANNELS (FTP)

7.1.7.1 FTP_ITC.1 TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The developer shall provide to the evaluator application layer configuration settings for all secure communication mechanisms specified by the FTP_ITC.1 requirement. This information should be sufficiently detailed to allow the evaluator to determine the application layer timeout settings for each

	cryptographic protocol. There is no expectation that this information must be recorded in any public-facing document or report. The evaluator shall perform the following tests: Test 1: The evaluators shall ensure that communications using each protocol with each authorized IT entity is tested during the course of the evaluation, setting up the connections as described in the guidance documentation and ensuring that communication is successful. Further assurance activities are associated with the specific protocols. For distributed TOEs the evaluator shall perform tests on all TOE components according to the mapping of external secure channels to TOE components in the Security Target. The developer shall provide to the evaluator application layer configuration settings for all secure communication mechanisms specified by the FTP_ITC.1 requirement. This information should be sufficiently detailed to allow the evaluator to determine the application layer timeout settings for each cryptographic protocol. There is no expectation that this information must be recorded in any public- facing document or report.
Test Steps	External Audit Server The testing for this is covered in FAU_STG_EXT.1 Test #1. <u>LDAP Server</u> • Configure the TOE to connect to the LDAP server. • Configure the LDAP server to allow authentication from the TOE. • Attempt to log into TOE using LDAP credentials to ensure it was successful. • Verify via Packet capture that the connection was established between TOE and the LDAP server. • Verify via audit logs that the connection was established between TOE and the LDAP server. The TOE is not distributed so there are no additional components to test.
Pass/Fail with Explanation	Pass. TOE successfully communicates to LDAP and Syslog server with an encrypted channel. The connection can be initiated by the TOE. TOE is not distributed TOE. This meets the testing requirements.

7.1.7.2 FTP_ITC.1 TEST #2 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The developer shall provide to the evaluator application layer configuration settings for all secure communication mechanisms specified by the FTP_ITC.1 requirement. This information should be sufficiently detailed to allow the evaluator to determine the application layer timeout settings for each cryptographic protocol. There is no expectation that this information must be recorded in any public-facing document or report. The evaluator shall perform the following tests: Test 2: 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. Further assurance activities are associated with the specific protocols. For distributed TOEs the evaluator shall perform tests on all TOE components according to the mapping of external secure channels to TOE components in the Security Target. The developer shall provide to the evaluator application layer configuration settings for all secure communication mechanisms specified by the FTP_ITC.1 requirement. This information should be sufficiently detailed to allow the evaluator to determine the application layer timeout settings for each cryptographic protocol. There is no expectation that this information must be recorded in any public- facing document or report.
Pass/Fail with Explanation	Pass. External Audit Server - testing for this is covered in FAU_STG_EXT.1 Test #1, we found that communication channel between the external Audit Server and TOE is initiated by TOE. LDAP Server - testing for this is covered in FTP_ITC.1 Test #1, we found that the communication channel between LDAP Server and TOE is initiated by TOE. TOE is not distributed TOE. This meets the testing requirements.

7.1.7.3 FTP_ITC.1 TEST #3 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The developer shall provide to the evaluator application layer configuration settings for all secure communication mechanisms specified by the FTP_ITC.1 requirement. This information should be sufficiently detailed to allow the evaluator to determine the application layer timeout settings for each

	cryptographic protocol. There is no expectation that this information must be recorded in any public-facing document or report. The evaluator shall perform the following tests: Test 3: The evaluator shall ensure, for each communication channel with an authorized IT entity, the channel data is not sent in plaintext. Further assurance activities are associated with the specific protocols. For distributed TOEs the evaluator shall perform tests on all TOE components according to the mapping of external secure channels to TOE components in the Security Target. The developer shall provide to the evaluator application layer configuration settings for all secure communication mechanisms specified by the FTP_ITC.1 requirement. This information should be sufficiently detailed to allow the evaluator to determine the application layer timeout settings for each cryptographic protocol. There is no expectation that this information must be recorded in any public- facing document or report.
Pass/Fail with Explanation	Pass. External Audit Server - testing for this is covered in FAU_STG_EXT.1 Test #1, we found that the communication channel between the external Audit Server and TOE channel data is not sent in plaintext. LDAP Server - testing for this is covered in FTP_ITC.1 Test #1, we found that the communication channel between the LDAP Server and TOE channel data is not sent in plaintext. TOE is not distributed TOE. This meets the testing requirements.

7.1.7.4 FTP_ITC.1 TEST #4 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The developer shall provide to the evaluator application layer configuration settings for all secure communication mechanisms specified by the FTP_ITC.1 requirement. This information should be sufficiently detailed to allow the evaluator to determine the application layer timeout settings for each cryptographic protocol. There is no expectation that this information must be recorded in any public-facing document or report. The evaluator shall perform the following tests:

	Test 4: 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, interrupt the connection of that IT entity for the following durations: i) a duration that exceeds the TOE's application layer timeout setting, ii) 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 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 TOE external interruption (such as a cable being physically removed or a virtual connection being disabled), another network device shall be used to interrupt the connection between the TOE and the distinct IT entity. The interruption shall be external to the TOE (i.e., by manipulating the test environment and not by TOE configuration change). Further assurance activities are associated with the specific protocols. For distributed TOEs the evaluator shall perform tests on all TOE components according to the mapping of external secure channels to TOE components in the Security Target. The developer shall provide to the evaluator application layer configuration settings for all secure communication mechanisms specified by the FTP_ITC.1 requirement. This information should be sufficiently detailed to allow the evaluator to determine the application layer timeout settings for each cryptographic protocol. There is no expectation that this information must be recorded in any public- facing document or report.
Test Steps	Syslog: • Initiate a connection between the TOE and Syslog server and ensure it is successful. • Verify the connection was established via logs. • Interrupt the connection between the TOE and Syslog server for a timeframe shorter (Less than 60 seconds) than the application layer timeout. • Verify via TOE logs the session is re-established between the TOE and Syslog server. • Verify via packet capture that no TSF data is sent in plaintext.

	• Initiate a connection between the TOE and Syslog server and ensure it is successful. • Verify the connection was established via logs. • Interrupt the connection between the TOE and Syslog server for a timeframe greater (more than 60 seconds) than the application layer timeout. • Verify via TOE logs the session is re-established between the TOE and Syslog server. • Verify via packet capture that no TSF data is sent in plaintext. LDAP: • Initiate a connection between the TOE and LDAP server by attempting user authentication. • Interrupt the connection between the TOE and LDAP server for a timeframe shorter (less than 60 seconds) than the application layer timeout. • Verify via packet capture that no TSF data is sent in plaintext. • Initiate a connection between the TOE and LDAP server by attempting user authentication. • Interrupt the connection between the TOE and LDAP server for a timeframe greater (more than 60 seconds) than the application layer timeout. • Verify via TOE logs the session is re-established between the TOE and LDAP server. • Verify via packet capture that no TSF data is sent in plaintext.
Pass/Fail with Explanation	Pass. The TOE ensures when the connectivity is restored after a connection outage with an external IT entity, communications are appropriately protected and no TSF data is sent in plaintext. TOE is not distributed TOE. This meets the testing requirements.

7.1.7.5 FTP_TRP.1/ADMIN TEST #1 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 1: 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. Further assurance activities are associated with the specific protocols.

	For distributed TOEs the evaluator shall perform tests on all TOE components according to the mapping of trusted paths to TOE components in the Security Target.
Test Steps	• Log into the TOE via WUI. • Verify audit logs that user was successfully logged in to the TOE. • Verify that the session was established and encrypted via packet capture.
Pass/Fail with Explanation	Pass. Remote administrative access to the TOE is over secured channels. TOE is not distributed TOE. This meets the testing requirements.

7.1.7.6 FTP_TRP.1/ADMIN TEST #2 (CPP_ND_V3.0E)

Item	Data
Test Assurance Activity	The evaluator shall perform the following tests: Test 2: The evaluator shall ensure, for each communication channel, the channel data is not sent in plaintext. Further assurance activities are associated with the specific protocols. For distributed TOEs the evaluator shall perform tests on all TOE components according to the mapping of trusted paths to TOE components in the Security Target.
Pass/Fail with Explanation	Pass. This test is performed in conjunction with FTP_TRP.1/Admin Test #1. Remote administrative access to the TOE is over secured channels and the data was not sent in plaintext. TOE is not distributed TOE. This meets the testing requirements.

8 CAVP ALGORITHM CERTIFICATE DETAILS

Each of these cryptographic algorithms have been validated as identified in the table below.

SFR	Algorithm Description	Implementation name	CAVP Alg.	CAVP Cert #	TOE Cards
FCS_CKM.1	ECC schemes using 'NIST curves' [**P-256, P-384, P-521**] that meet the following: FIPS PUB 186-5, "Digital Signature Standard (DSS)", Appendix A.2.	Progress LoadMaster Cryptographic Library Version 7.2	ECDSA KeyGen (FIPS186-5) (Curve P-256, P-384, P-521) ECDSA KeyVer (FIPS186-5) (Curve P-256, P-384, P-521)	A7300	LoadMaster X25-NG LoadMaster X40-NG ECS CM H2 NG ECS CM H3 NG Virtual LoadMaster
FCS_CKM.2	Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography"	Progress LoadMaster Cryptographic Library Version 7.2	KAS-ECC-SSC Sp800-56Ar3 (Domain Parameter Generation Methods: P-256, P-384, P-521)	A7300	LoadMaster X25-NG LoadMaster X40-NG ECS CM H2 NG ECS CM H3 NG Virtual LoadMaster
FCS_COP.1/ DataEncryption	GCM *mode* and cryptographic key sizes 128 bits, 256 bits that meet: *AES as specified in ISO 18033-3 and* GCM as specified in ISO 19772	Progress LoadMaster Cryptographic Library Version 7.2	AES-GCM (Key Length: 128, 256)	A7300	LoadMaster X25-NG LoadMaster X40-NG ECS CM H2 NG ECS CM H3 NG Virtual LoadMaster
	CBC *mode* and cryptographic key sizes 128 bits, 256 bits that meet: *AES as specified in ISO 18033-3,* CBC as specified in ISO 10116.		AES-CBC (Key Length: 128, 256)		

SFR	Algorithm Description	Implementation name	CAVP Alg.	CAVP Cert #	TOE Cards
	CTR *mode* and cryptographic key sizes 256 bits that meet: *AES as specified in ISO 18033-3,* CTR as specified in ISO 10116.		AES-CTR (Key Length: 256)		
FCS_COP.1/ SigGen	For RSA schemes: FIPS PUB 186-5, "Digital Signature Standard (DSS)", Section 5.4, using PKCS #1 v2.2 Signature Schemes RSASSA-PSS and/or RSASSA-PKCS1v1_5;	Progress LoadMaster Cryptographic Library Version 7.2	RSA SigGen (FIPS186-5) (Modulo 2048) RSA SigVer (FIPS186-5) (Modulo 2048)	A7300	LoadMaster X25-NG LoadMaster X40-NG ECS CM H2 NG ECS CM H3 NG Virtual LoadMaster
	For ECDSA schemes implementing [P-256, P-384, P-521] curves that meet the following : FIPS PUB 186-5, "Digital Signature Standard (DSS)", Section 6 and NIST SP 800-186 Section 3.2.1, Implementing Weierstrass curves;	Progress LoadMaster Cryptographic Library Version 7.2	ECDSA SigGen (FIPS186-5) (Curve P-256, P-384, P-521) ECDSA SigVer (FIPS186-5) (Curve P-256, P-384, P-521)	A7300	LoadMaster X25-NG LoadMaster X40-NG ECS CM H2 NG ECS CM H3 NG Virtual LoadMaster
FCS_COP.1/ Hash	[**SHA-1, SHA-256, SHA-384, SHA-512**] and message digest sizes [**160, 256, 384, 512**] bits	Progress LoadMaster Cryptographic Library Version 7.2	SHA-1, SHA-256, SHA-384, SHA-512	A7300	LoadMaster X25-NG LoadMaster X40-NG ECS CM H2 NG ECS CM H3 NG Virtual LoadMaster

SFR	Algorithm Description	Implementation name	CAVP Alg.	CAVP Cert #	TOE Cards
FCS_COP.1/ KeyedHash	[**HMAC-SHA-1, HMAC-SHA- 256, HMAC-SHA-384**] and cryptographic key sizes [**160-bits, 256-bits, 384 bits**] and message digest sizes [**160, 256, 384**] bits	Progress LoadMaster Cryptographic Library Version 7.2	HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384	A7300	LoadMaster X25-NG LoadMaster X40-NG ECS CM H2 NG ECS CM H3 NG Virtual LoadMaster
FCS_RBG_EXT.1	CTR_DRBG (AES) in accordance with ISO/IEC 18031:2011 with a minimum of 256-bits	Progress LoadMaster Cryptographic LibraryProgress LoadMaster Cryptographic Library Version 7.2	Counter DRBG (Mode: AES-256)	A7300	LoadMaster X25-NG LoadMaster X40-NG ECS CM H2 NG ECS CM H3 NG Virtual LoadMaster

9 CONCLUSION

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