Progress LoadMaster Security Target

Document Version: **1.7**
Date: **June 18, 2026**

intertek acumen security

2400 Research Blvd
Suite 395
Rockville, MD 20850

Revision History

Version	Date	Changes
Version 1.0	April 22, 2025	Initial Release
Version 1.1	May 30, 2025	Updated as per the responses from the vendor and TLS testing observations.
Version 1.2	June 16, 2025	Updated as per the internal comments.
Version 1.3	October 10, 2025	Updated to remove TLS 1.3.
Version 1.4	November 21, 2025	Minor updates prior to check-in submission.
Version 1.5	February 23, 2026	Internal review comments.
Version 1.6	April 8, 2026	Addressing Validator ECR comments
Version 1.7	June 18,2026	Addressing Validator ECR comments

Contents

1. INTRODUCTION

The Security Target (ST) serves as the basis for the Common Criteria (CC) evaluation and identifies the Target of Evaluation (TOE), the scope of the evaluation, and the assumptions made throughout. This document will also describe the intended operational environment of the TOE, and the functional and assurance requirements that the TOE meets.

1.1 SECURITY TARGET AND TOE REFERENCE

This section provides the information needed to identify and control the TOE and the ST.

Table 1 – TOE/ST Identification

Category	Identifier
ST Title	Progress LoadMaster Security Target
ST Version	1.7
ST Date	June 18, 2026
ST Author	Intertek Acumen Security
TOE Identifier	Progress LoadMaster
TOE Version	LoadMaster OS 7.2.54.18
TOE Models	LoadMaster X25-NG, LoadMaster X40-NG, ECS CM H2 NG, ECS CM H3 NG and Virtual LoadMaster
TOE Developer	Progress Software Corporation
Key Words	Network Device, Load Balancer

1.2 TOE OVERVIEW

The TOE is the Progress Software Corporation's Progress LoadMaster LoadMaster X25-NG, LoadMaster X40-NG, ECS CM H2 NG, ECS CM H3 NG and Virtual LoadMaster 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. The evaluated functionality is described in Section 1.3.2 below.

1.3 TOE DESCRIPTION

Progress LoadMaster is a load balancer and application delivery controller (ADC) appliance. It is designed to increase the availability, scalability, and security of applications and data centers. The TOE suits all sizes of organizations with integrated hardware acceleration and support for up to 35 million concurrent connections. The TOE is a network device.

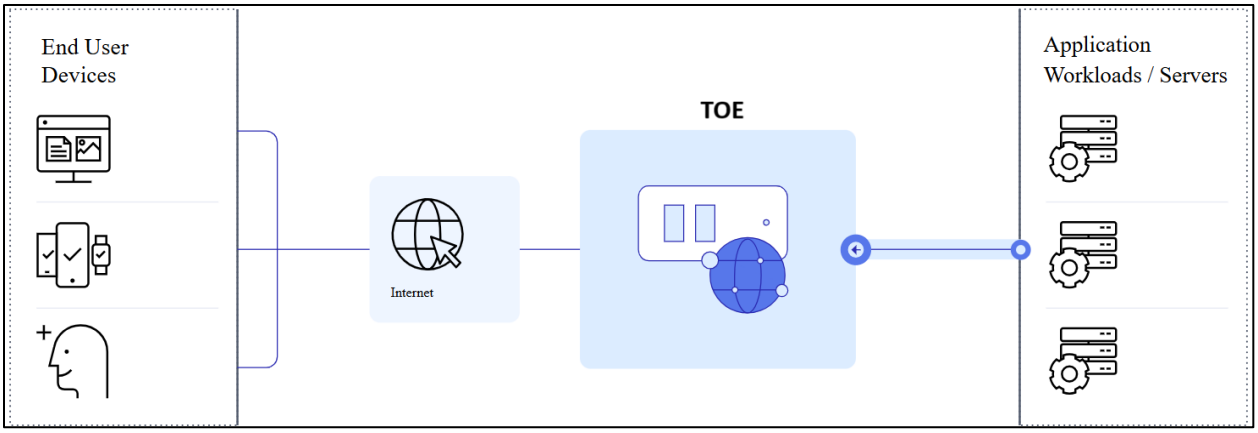

Figure 1 – Representative TOE Deployment

1.3.1 PHYSICAL BOUNDARIES

The TOE boundary consists of one of the appliances listed below. The LoadMaster X25-NG, LoadMaster X40-NG, ECS CM H2 NG and ECS CM H3 NG are physical devices while the Virtual LoadMaster is a virtual machine which runs on ESXi. The virtual TOE is conformant with Case 1 as described in the NDcPP:

Table 2 – TOE Physical Boundary Components

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	2 1Gb Ethernet 8 10Gb Ethernet Fiber 4 25Gb Ethernet Fiber	2 10Gb virtual NIC (evaluated)

Model	LoadMaster X25-NG	LoadMaster X40-NG	ECS CM H2 NG	ECS CM H3 NG	Virtual LoadMaster
			4 100Gb Ethernet Fiber	4 100Gb Ethernet Fiber	
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

1.3.2 SECURITY FUNCTIONS PROVIDED BY THE TOE

The TOE provides the security functions required by the Collaborative Protection Profile for Network Devices, hereafter referred to as NDcPP v3.0e or NDcPP.

1.3.2.1 SECURITY AUDIT

The TOE generates audit records for security relevant events. The audit events are associated with the administrator or processes. The audit records are transmitted over TLS to an external audit server.

1.3.2.2 CRYPTOGRAPHIC SUPPORT

The TOE provides the following cryptographic services described below.

Table 3– Cryptographic Services

Service	Use
TLS Client	Secure connection to remote syslog servers.
TLS Client	Secure connection to remote LDAP server.
TLS/HTTPS Server	Secures connections with remote administrators.
Verification of Updates	Digital signature verification prior to installing an update.

Each of these cryptographic algorithms have been validated for conformance to the requirements specified in their respective standards, as identified below.

1.3.2.3 IDENTIFICATION AND AUTHENTICATION

The TOE supports a password-based authentication mechanism which automatically locks users after a pre-configured number of failed attempts. The TOE also validates X.509 certificates in support of TLS.

1.3.2.4 SECURITY MANAGEMENT

The TOE provides management capabilities via Console and a Web-based GUI, accessed over HTTPS. Management functions allow the administrators to configure the system, install updates, and manage users.

1.3.2.5 PROTECTION OF THE TSF

The TOE prevents the reading of plaintext passwords and keys. The TOE provides a reliable timestamp for its own use. The reliable timestamp can be set by a security administrator or authenticated NTP. To protect the integrity of its security functions, the TOE implements a suite of self-tests at startup and halts or disables affected functionality if a self-test fails. The TOE ensures that updates to the TOE are authenticated by verifying a digital signature prior to installing any update.

1.3.2.6 TOE ACCESS

The TOE monitors local and remote administrative sessions for inactivity and either locks or terminates the session when a threshold time period is reached. An advisory notice is displayed at the start of each session.

1.3.2.7 TRUSTED PATH/CHANNELS

The TOE initiates a TLS trusted channel with a syslog server and LDAP authentication server (as configured).

The TOE is a TLS/HTTPS server that allows remote administrators to establish a trusted path with the TOE.

1.3.3 TOE DOCUMENTATION

The following documents are essential to understanding and controlling the TOE in the evaluated configuration:

- Progress LoadMaster OS 7.2.54.18 Common Criteria Conformance Guide, v0.5

1.4 TOE ENVIRONMENT

The following environmental components are required to operate the TOE in the evaluated configuration:

- 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.

TOE Management interfaces for Physical TOE are shown below,

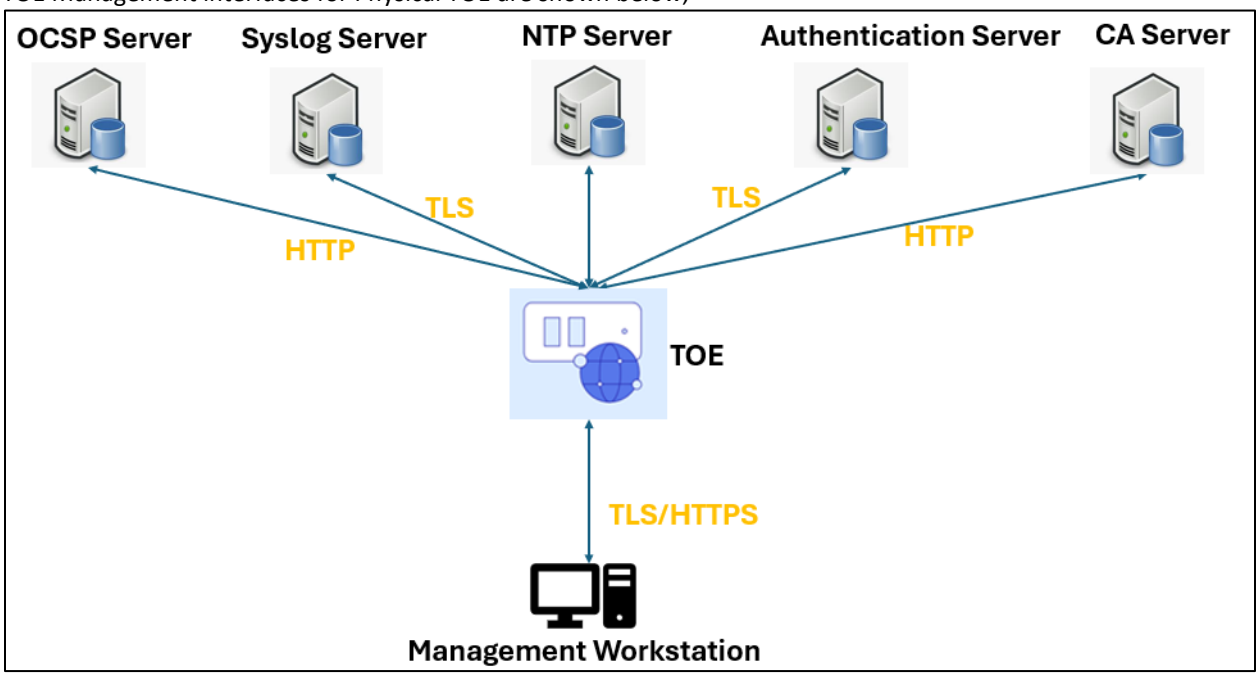

Figure 2 – Physical TOE Management Interfaces

TOE Management interfaces for Virtual TOE are shown below,

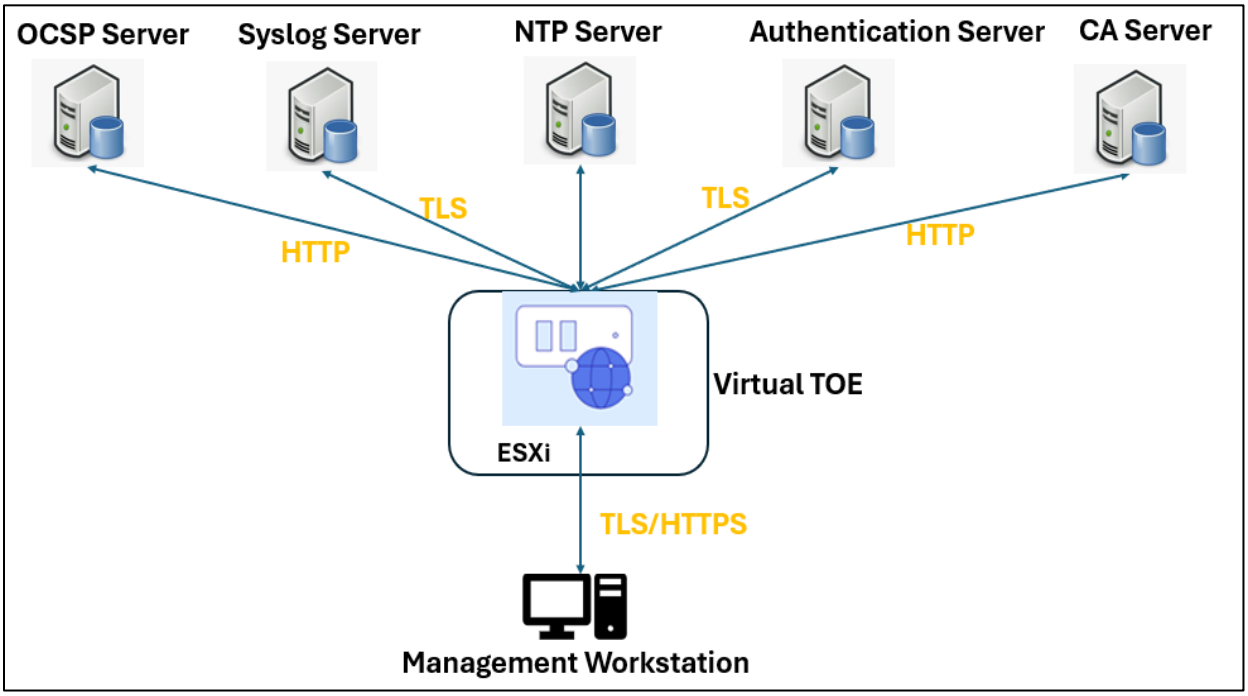

Figure 3 – Virtual TOE Management Interfaces

1.5 PRODUCT FUNCTIONALITY NOT INCLUDED IN THE SCOPE OF THE EVALUATION

The following product functionality is not included in the CC evaluation:

- SSH
- Management API
- IPv6
- RADIUS Server

2. CONFORMANCE CLAIMS

This section identifies the TOE conformance claims, conformance rationale, and relevant Technical Decisions (TDs).

2.1 CC CONFORMANCE CLAIMS

The TOE is conformant to the following:

- Common Criteria for Information Technology Security Evaluations Part 1, Version 3.1, Revision 5, April 2017
- Common Criteria for Information Technology Security Evaluations Part 2, Version 3.1, Revision 5, April 2017 (Extended)
- Common Criteria for Information Technology Security Evaluations Part 3, Version 3.1, Revision 5, April 2017 (Conformant)

2.2 PROTECTION PROFILE CONFORMANCE

This ST claims exact conformance to the following:

- collaborative Protection Profile for Network Devices, Version 3.0e, (CPP_ND_V3.0E)

2.3 CONFORMANCE RATIONALE

This ST provides exact conformance to the items listed in the previous section. The security problem definition, security objectives, and security requirements in this ST are all taken from the Protection Profile (PP), performing only the operations defined there.

2.3.1 TECHNICAL DECISIONS

All NIAP TDs issued to date and applicable to NDcPP v3.0e have been considered. Table 4 identifies all applicable TDs.

Table 4 – Relevant Technical Decisions

Technical Decision	Applicable (Y/N)	Exclusion Rationale (if applicable)
TD1033 – Sunset Dates for NDcPP Configurations	Y	
TD0990 – NIT Technical Decision: CTR_DRBG in FCS_RBG_EXT.1.2	Y	
TD0973 – NIT Technical Decision: FCS_(D)TLSS_EXT.1.3 Test 2 DHE Ciphersuite Conditionality	Y	

Technical Decision	Applicable (Y/N)	Exclusion Rationale (if applicable)
TD0923 - NIT Technical Decision: Auditable event for FAU_STG_EXT.1 in FAU_GEN.1.2	Y	
TD0921 - NIT Technical Decision: Addition of FIPS PUB 186-5 and Correction of Assignment	Y	
TD0900 - NIT Technical Decision: Clarification to Local Administrator Access in FIA_UIA_EXT.1.3	Y	
TD0899 - NIT Technical Decision: Correction of Renegotiation Test for TLS 1.2	Y	
TD0886 - Clarification to FAU_STG_EXT.1 Test 6	Y	
TD0880 - NIT Decision: Removal of Duplicate Selection in FMT_SMF.1.1	Y	
TD0879 - NIT Decision: Correction of Chapter Headings in CPP_ND_V3.0E	Y	
TD0868 - NIT Technical Decision: Clarification of time frames in FCS_IPSEC_EXT.1.7 and FCS_IPSEC_EXT.1.8	N	This TD addresses IPsec functionality, and this TOE does not include IPsec functionality.
TD0836 - NIT Technical Decision: Redundant Requirements in FPT_TST_EXT.1	Y	

3. SECURITY PROBLEM DEFINITION

The security problem definition has been taken directly from the NDcPP specified in Section 2.2 and is reproduced here for the convenience of the reader. The security problem is described in terms of the threats that the TOE is expected to address, assumptions about the operational environment, and any Organizational Security Policies (OSPs) that the TOE is expected to enforce.

3.1 THREATS

The threats included in Table 5 are drawn directly from the NDcPP specified in Section 2.2.

Table 5 – Threats

ID	Threat
T.UNAUTHORIZED_ADMINISTRATOR_ACCESS	Threat agents may attempt to gain Administrator access to the Network Device by nefarious means such as masquerading as an Administrator to the device, masquerading as the device to an Administrator, replaying an administrative session (in its entirety, or selected portions), or performing man-in-the-middle attacks, which would provide access to the administrative session, or sessions between Network Devices. Successfully gaining Administrator access allows malicious actions that compromise the security functionality of the device and the network on which it resides.
T.WEAK_CRYPTOGRAPHY	Threat agents may exploit weak cryptographic algorithms or perform a cryptographic exhaust against the key space. Poorly chosen encryption algorithms, modes, and key sizes will allow attackers to compromise the algorithms, or brute force exhaust the key space and give them unauthorized access allowing them to read, manipulate and/or control the traffic with minimal effort.
T.UNTRUSTED_COMMUNICATION_CHANNELS	Threat agents may attempt to target Network Devices that do not use standardized secure tunnelling protocols to protect the critical network traffic. Attackers may take advantage of poorly designed protocols or poor key management to successfully perform man-in-the-middle attacks, replay attacks, etc. Successful attacks will result in loss of confidentiality and integrity of the critical network traffic, and potentially could lead to a compromise of the Network Device itself.
T.WEAK_AUTHENTICATION_ENDPOINTS	Threat agents may take advantage of secure protocols that use weak methods to authenticate the endpoints, e.g. a shared password that is guessable or transported as plaintext. The consequences are the same as a poorly designed protocol, the attacker could masquerade as the Administrator or another device, and the attacker could insert themselves into the network stream and perform a man-in-the-middle attack. The result is the critical

ID	Threat
	network traffic is exposed and there could be a loss of confidentiality and integrity, and potentially the Network Device itself could be compromised.
T.UPDATE_COMPROMISE	Threat agents may attempt to provide a compromised update of the software or firmware which undermines the security functionality of the device. Non-validated updates or updates validated using non-secure or weak cryptography leave the update firmware vulnerable to surreptitious alteration.
T.UNDETECTED_ACTIVITY	Threat agents may attempt to access, change, and/or modify the security functionality of the Network Device without Administrator awareness. This could result in the attacker finding an avenue (e.g., misconfiguration, flaw in the product) to compromise the device and the Administrator would have no knowledge that the device has been compromised.
T.SECURITY_FUNCTIONALITY_COMPROMISE	Threat agents may compromise credentials and device data enabling continued access to the Network Device and its critical data. The compromise of credentials includes replacing existing credentials with an attacker's credentials, modifying existing credentials, or obtaining the Administrator or device credentials for use by the attacker. Threat agents may also be able to take advantage of weak administrative passwords to gain privileged access to the device.
T.SECURITY_FUNCTIONALITY_FAILURE	An external, unauthorized entity could make use of failed or compromised security functionality and might therefore subsequently use or abuse security functions without prior authentication to access, change or modify device data, critical network traffic or security functionality of the device.

3.2 ASSUMPTIONS

The assumptions included in Table 6 are drawn directly from NDcPP.

Table 6 – Assumptions

ID	Assumption
A.PHYSICAL_PROTECTION	The Network Device is assumed to be physically protected in its operational environment and not subject to physical attacks that compromise the security or interfere with the device's physical interconnections and correct operation. This protection is assumed to be sufficient to protect the device and the data it contains. As a result, the cPP does not include any requirements on physical tamper protection or other physical attack mitigations. The cPP does not expect the product to defend against physical access to the device that allows unauthorized entities to extract data, bypass other controls, or otherwise manipulate the device. For vNDs, this assumption applies to the physical platform on which the VM runs.
A.LIMITED_FUNCTIONALITY	The device is assumed to provide networking functionality as its core function and not provide functionality/services that could be deemed as general purpose computing. For example, the device should not provide a computing platform for general purpose applications (unrelated to networking functionality). If a virtual TOE evaluated as a pND, following Case 2 vNDs as specified in Section 1.2, the VS is considered part of the TOE with only one vND instance for each physical hardware platform. The exception being where components of a distributed TOE run inside more than one virtual machine (VM) on a single VS. In Case 2 vND, no non-TOE guest VMs are allowed on the platform
A.NO_THRU_TRAFFIC_PROTECTION	A standard/generic Network Device does not provide any assurance regarding the protection of traffic that traverses it. The intent is for the Network Device to protect data that originates on or is destined to the device itself, to include administrative data and audit data. Traffic that is traversing the Network Device, destined for another network entity, is not covered by the ND cPP. It is assumed that this protection will be covered by cPPs and PP-Modules for particular types of Network Devices (e.g., firewall).

ID	Assumption
A.TRUSTED_ADMINISTRATOR	The Security Administrator(s) for the Network Device are assumed to be trusted and to act in the best interest of security for the organization. This includes appropriately trained, following policy, and adhering to guidance documentation. Administrators are trusted to ensure passwords/credentials have sufficient strength and entropy and to lack malicious intent when administering the device. The Network Device is not expected to be capable of defending against a malicious Administrator that actively works to bypass or compromise the security of the device. For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are expected to fully validate (e.g. offline verification) any CA certificate (root CA certificate or intermediate CA certificate) loaded into the TOE's trust store (aka 'root store', ' trusted CA Key Store', or similar) as a trust anchor prior to use (e.g. offline verification).
A.REGULAR_UPDATES	The Network Device firmware and software is assumed to be updated by an Administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
A.ADMIN_CREDENTIALS_SECURE	The Administrator's credentials (private key) used to access the Network Device are protected by the platform on which they reside.
A.RESIDUAL_INFORMATION	The Administrator must ensure that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment.
A.VS_TRUSTED_ADMINISTRATOR	The Security Administrators for the VS are assumed to be trusted and to act in the best interest of security for the organization. This includes not interfering with the correct operation of the device. The Network Device is not expected to be capable of defending against a malicious VS Administrator that actively works to bypass or compromise the security of the device.
A.VS_REGULAR_UPDATES	The VS software is assumed to be updated by the VS Administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
A.VS_ISOLATION	For vNDs, it is assumed that the VS provides, and is configured to provide sufficient isolation between software running in VMs on the same physical platform. Furthermore, it is assumed that the VS adequately protects itself from software running inside VMs on the same physical platform.
A.VS_CORRECT_CONFIGURATION	For vNDs, it is assumed that the VS and VMs are correctly configured to support ND functionality implemented in VMs.

3.3 ORGANIZATIONAL SECURITY POLICIES

The OSPs included in Table 7 are drawn directly from the NDcPP.

Table 7 – OSPs

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

4. SECURITY OBJECTIVES

The security objectives have been taken directly from the NDcPP and are reproduced here for the convenience of the reader.

4.1 SECURITY OBJECTIVES FOR THE OPERATIONAL ENVIRONMENT

Security objectives for the operational environment assist the TOE in correctly providing its security functionality. These objectives, which are found in the table below, track with the assumptions about the TOE operational environment.

Table 8 – Security Objectives for the Operational Environment

ID	Objectives for the Operational Environment
OE.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.
OE.NO_GENERAL_PURPOSE	There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE. Note: For vNDs the TOE includes only the contents of the its own VM, and does not include other VMs or the VS.
OE.NO_THRU_TRAFFIC_PROTECTION	The TOE does not provide any protection of traffic that traverses it. It is assumed that protection of this traffic will be covered by other security and assurance measures in the operational environment.
OE.TRUSTED_ADMIN	Security Administrators are trusted to follow and apply all guidance documentation in a trusted manner. For vNDs, this includes the VS Administrator responsible for configuring the VMs that implement ND functionality. For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are assumed to monitor the revocation status of all certificates in the TOE's trust store and to remove any certificate from the TOE's trust store in case such certificate can no longer be trusted.
OE.UPDATES	The TOE firmware and software is updated by an Administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
OE.ADMIN_CREDENTIALS_SECURE	The Administrator's credentials (private key) used to access the TOE must be protected on any other platform on which they reside.

ID	Objectives for the Operational Environment
OE.RESIDUAL_INFORMATION	The Security Administrator ensures that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment. For vNDs, this applies when the physical platform on which the VM runs is removed from its operational environment.
OE.VM_CONFIGURATION	For vNDs, the Security Administrator ensures that the VS and VMs are configured to • reduce the attack surface of VMs as much as possible while supporting ND functionality (e.g., remove unnecessary virtual hardware, turn off unused inter-VM communications mechanisms), and • correctly implement ND functionality (e.g., ensure virtual networking is properly configured to support network traffic, management channels, and audit reporting). The VS should be operated in a manner that reduces the likelihood that vND operations are adversely affected by virtualisation features such as cloning, save/restore, suspend/resume, and live migration. If possible, the VS should be configured to make use of features that leverage the VS's privileged position to provide additional security functionality. Such features could include malware detection through VM introspection, measured VM boot, or VM snapshot for forensic analysis.

5. SECURITY REQUIREMENTS

This section identifies the Security Functional Requirements (SFRs) for the TOE. The SFRs included in this section are derived from Part 2 of the Common Criteria for Information Technology Security Evaluation, Version 3.1, Revisions 5, April 2017, and all international interpretations.

Table 9 – SFRs

Requirement	Description
FAU_GEN.1	Audit Data Generation
FAU_GEN.2	User Identity Association
FAU_STG_EXT.1	Protected Audit Event Storage
FAU_STG_EXT.3	Action in Case of Possible Audit Data Loss
FCS_CKM.1	Cryptographic Key Generation
FCS_CKM.2	Cryptographic Key Establishment
FCS_CKM.4	Cryptographic Key Destruction
FCS_COP.1/DataEncryption	Cryptographic Operation (AES Data Encryption/Decryption)
FCS_COP.1/SigGen	Cryptographic Operation (Signature Generation and Verification)
FCS_COP.1/Hash	Cryptographic Operation (Hash Algorithm)
FCS_COP.1/KeyedHash	Cryptographic Operation (Keyed Hash Algorithm)
FCS_HTTPS_EXT.1	HTTPS Protocol
FCS_NTP_EXT.1	NTP Protocol
FCS_RBG_EXT.1	Random Bit Generation
FCS_TLSC_EXT.1	TLS Client Protocol
FCS_TLSS_EXT.1	TLS Server Protocol
FIA_AFL.1	Authentication Failure Handling
FIA_PMG_EXT.1	Password Management
FIA_UIA_EXT.1	User Identification and Authentication
FIA_UAU.7	Protected Authentication Feedback (Refinement)
FIA_X509_EXT.1/Rev	X.509 Certificate Validation
FIA_X509_EXT.2	X.509 Certificate Authentication
FIA_X509_EXT.3	X.509 Certificate Requests
FMT_MOF.1/ManualUpdate	Management of Security Functions Behaviour
FMT_MOF.1/Services	Management of Security Functions Behaviour
FMT_MOF.1/Functions	Management of Security Functions Behaviour
FMT_MTD.1/CoreData	Management of TSF Data
FMT_MTD.1/CryptoKeys	Management of TSF Data
FMT_SMF.1	Specification of Management Functions
FMT_SMR.2	Restrictions on Security Roles

Requirement	Description
FPT_SKP_EXT.1	Protection of TSF Data (for reading of all symmetric keys)
FPT_APW_EXT.1	Protection of Administrator Passwords
FPT_TST_EXT.1	TSF Testing
FPT_STM_EXT.1	Reliable Time Stamps
FPT_TUD_EXT.1	Trusted Update
FTA_SSL.3	TSF-initiated Termination
FTA_SSL.4	User-initiated Termination
FTA_SSL_EXT.1	TSF-initiated Session Locking
FTA_TAB.1	Default TOE Access Banner
FTP_ITC.1	Inter-TSF Trusted Channel
FTP_TRP.1/Admin	Trusted Path

5.1 CONVENTIONS

The CC allows the following types of operations to be performed on the functional requirements: assignments, selections, refinements, and iterations. The following font conventions are used within this document to identify operations defined by CC:

- Assignment: Indicated with *italicized* text;
- Refinement: Indicated with **bold** text and ~~strikethroughs~~;
- Selection: Indicated with underlined text;
- Iteration: Indicated by appending the iteration identifier after a slash, e.g., /SigGen.
- Where operations were completed in the PP and relevant EPs/Modules/Packages, the formatting used in the PP has been retained.
- Extended SFRs are identified by the addition of "EXT" after the requirement name.

5.2 SECURITY FUNCTIONAL REQUIREMENTS

This section includes the security functional requirements for this ST.

5.2.1 SECURITY AUDIT (FAU)

5.2.1.1 FAU_GEN.1 AUDIT DATA GENERATION

FAU_GEN.1.1

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

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

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

c) *All administrative actions comprising:*

- *Administrative login and logout (name of Administrator account shall be logged if individual accounts are required for Administrators).*

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

d) Specifically defined auditable events listed in Table 10.

FAU_GEN.1.2

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

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

b) For each audit event type, based on the auditable event definitions of the functional components included in the cPP/ST, *information specified in column three of Table 10.*

Table 10 – Security Functional Requirements and Auditable Events

Requirement	Auditable Events	Additional Audit Record Contents
FAU_GEN.1	None.	None.
FAU_GEN.2	None.	None.
FAU_STG_EXT.1	Configuration of local audit settings.	Identity of account making changes to the audit configuration.
FAU_STG_EXT.3	Low storage space for audit events.	None.
FCS_CKM.1	None.	None.
FCS_CKM.2	None.	None.
FCS_CKM.4	None.	None.
FCS_COP.1/DataEncryption	None.	None.
FCS_COP.1/SigGen	None.	None.
FCS_COP.1/Hash	None.	None.
FCS_COP.1/KeyedHash	None.	None.
FCS_HTTPS_EXT.1	Failure to establish a HTTPS Session.	Reason for failure
FCS_NTP_EXT.1	• Configuration of a new time server • Removal of configured time server	Identity if new/removed time server
FCS_RBG_EXT.1	None.	None.
FCS_TLSC_EXT.1	Failure to establish a TLS Session	Reason for failure
FCS_TLSS_EXT.1	Failure to establish a TLS Session	Reason for failure
FIA_AFL.1	Unsuccessful login attempts limit is met or exceeded.	Origin of the attempt (e.g., IP address).

Requirement	Auditable Events	Additional Audit Record Contents
FIA_PMG_EXT.1	None.	None.
FIA_UIA_EXT.1	All use of identification and authentication mechanisms.	Origin of the attempt (e.g., IP address).
FIA_UAU.7	None.	None.
FIA_X509_EXT.1/Rev	• Unsuccessful attempt to validate a certificate • Any addition, replacement or removal of trust anchors in the TOE's trust store	• Reason for failure of certificate validation • Identification of certificates added, replaced or removed as trust anchor in the TOE's trust store
FIA_X509_EXT.2	None	None
FIA_X509_EXT.3	None.	None.
FMT_MOF.1/ManualUpdate	Any attempt to initiate a manual update.	None.
FMT_MOF.1/Services	None.	None.
FMT_MOF.1/Functions	None.	None.
FMT_MTD.1/CoreData	None.	None.
FMT_MTD.1/CryptoKeys	None.	None.
FMT_SMF.1	All management activities of TSF data.	None.
FMT_SMR.2	None.	None.
FPT_APW_EXT.1	None.	None.
FPT_SKP_EXT.1	None.	None.
FPT_TST_EXT.1	None.	None.
FPT_STM_EXT.1	Discontinuous changes to time - either Administrator actuated or changed via an automated process. (Note that no continuous changes to time need to be logged. See also application note on FPT_STM_EXT.1)	For discontinuous changes to time: The old and new values for the time. Origin of the attempt to change time for success and failure (e.g., IP address).
FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure)	None.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	None.
FTA_SSL.4	The termination of an interactive session.	None.

Requirement	Auditable Events	Additional Audit Record Contents
FTA_SSL_EXT.1 (if "lock the session" is selected)	Any attempts at unlocking of an interactive session.	None.
FTA_SSL_EXT.1 (if "terminate the session" is selected)	The termination of a local session by the session lock.	None.
FTA_TAB.1	None.	None.
FTP_ITC.1	• Initiation of the trusted channel. • Termination of the trusted channel. • Failure of the trusted channel functions.	• None • None • Reason for failure
FTP_TRP.1/Admin	• Initiation of the trusted path. • Termination of the trusted path. • Failure of the trusted path functions.	• None • None • Reason for failure

5.2.1.2 FAU_GEN.2 USER IDENTITY ASSOCIATION

FAU_GEN.2.1
For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.

5.2.1.3 FAU_STG_EXT.1 PROTECTED AUDIT EVENT STORAGE

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

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

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

].

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

FAU_STG_EXT.1.4
The TSF shall be able to store [persistent] audit records locally with a minimum storage size of [*7GB*].

FAU_STG_EXT.1.5
The TSF shall [drop new audit data] when the local storage space for audit data is full.

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

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

FAU_STG_EXT.3.1
The TSF shall generate a warning to inform the Administrator before the audit trail exceeds the local audit trail storage capacity.

5.2.2 CRYPTOGRAPHIC SUPPORT (FCS)

5.2.2.1 FCS_CKM.1 CRYPTOGRAPHIC KEY GENERATION

FCS_CKM.1.1 [TD0921]
The TSF shall generate **asymmetric** cryptographic key in accordance with a specified cryptographic key generation algorithm: [

- ECC schemes using 'NIST curves' [P-256, P-384, P-521] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4, or FIPS PUB 186-5, "Digital Signature Standard (DSS)", Appendix A.2, or ISO/IEC 14888-3, "IT Security techniques – Digital signatures with appendix - Part 3: Discrete logarithm based mechanisms", Section 6.6.

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

5.2.2.2 FCS_CKM.2 CRYPTOGRAPHIC KEY ESTABLISHMENT

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

- 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";.

] ~~that meets the following: [assignment: *list of standards*].~~

5.2.2.3 FCS_CKM.4 CRYPTOGRAPHIC KEY DESTRUCTION

FCS_CKM.4.1

The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method

- *For plaintext keys in volatile storage, the destruction shall be executed by a* [single overwrite consisting of [zeroes], destruction of reference to the key directly followed by a request for garbage collection]*;*
- *For plaintext keys in non-volatile storage, the destruction shall be executed by the invocation of an interface provided by a part of the TSF that* [
    - logically addresses the storage location of the key and performs a [single] overwrite consisting of [zeroes];]

that meets the following: *No Standard.*

5.2.2.4 FCS_COP.1/DATAENCRYPTION CRYPTOGRAPHIC OPERATIONS (AES DATA ENCRYPTION/DECRYPTION)

FCS_COP.1.1/DataEncryption

The TSF shall perform *encryption/decryption* in accordance with a specified cryptographic algorithm *AES used in* [CBC, CTR, GCM] *mode* and cryptographic key sizes [128 bits, 256 bits] that meet the following: *AES as specified in ISO 18033-3,* [CBC as specified in ISO 10116, CTR as specified in ISO 10116, GCM as specified in ISO 19772].

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

FCS_COP.1.1/SigGen [TD0921]

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

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

]

and cryptographic key sizes [

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

]

that meets the following: [

- For RSA schemes: FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Section 5.5 using PKCS #1 v2.1 or FIPS PUB 186-5, “Digital Signature Standard (DSS)”, Section 5.4, using PKCS #1 v2.2 Signature Schemes RSASSA-PSS and/or RSASSA-PKCS1v1_5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3,
- For ECDSA schemes implementing [P-256, P-384, P-521] curves that meet the following : FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Section 6 and Appendix D, Implementing “NIST Recommended” curves; or FIPS PUB 186-5, “Digital Signature Standard (DSS)”, Section 6 and NIST SP 800-186 Section 3.2.1, Implementing Weierstrass curves; or ISO/IEC 14888-3, “IT Security techniques - Digital signatures with appendix - Part 3: Discrete logarithm based mechanisms”, Section 6.6.

].

5.2.2.6 FCS_COP.1/HASH CRYPTOGRAPHIC OPERATIONS (HASH ALGORITHM)

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

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

FCS_COP.1.1/KeyedHash
The TSF shall perform *keyed-hash message authentication* in accordance with a specified cryptographic algorithm [HMAC-SHA-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*** that meet the following: *ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2"*.

5.2.2.8 FCS_HTTPS_EXT.1 HTTPS PROTOCOL

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

FCS_HTTPS_EXT.1.2
The TSF shall implement the HTTPS ~~protocol~~ using TLS.

5.2.2.9 FCS_NTP_EXT.1 NTP PROTOCOL

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

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

- Authentication using [SHA1] as the message digest algorithm(s);

].

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

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

5.2.2.10 FCS_RBG_EXT.1 RANDOM BIT GENERATION

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

FCS_RBG_EXT.1.2 [TD0990]
The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [*[One]* platform-based noise source] with a minimum of [*256 bits*] of entropy at least equal to

[

- the greatest security strength, according to ISO/IEC 18031:2011 Table C.1 "Security Strength Table for Hash Functions",

].

5.2.2.11 FCS_TLSC_EXT.1 TLS CLIENT PROTOCOL

FCS_TLSC_EXT.1.1
The TSF shall implement [TLS 1.2 (RFC 5246)] supporting 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

] and no other ciphersuites.

FCS_TLSC_EXT.1.2
The TSF shall verify that the presented identifier matches [the reference identifier per RFC 6125 Section 6, IPv4 address in the CN or in the SAN and no other attribute types].

FCS_TLSC_EXT.1.3
The TSF shall not establish a trusted channel if the server certificate is invalid [

- without any administrator override mechanism.

].

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

FCS_TLSC_EXT.1.5
The TSF shall [

- present the signature_algorithms extension with support for the following algorithms:
  [
  - ecdsa_secp256r1 with sha256(0x0403),
  - ecdsa_secp384r1 with sha384(0x0503),
  - ecdsa_secp521r1 with sha512(0x0603),

  ] and no other algorithms;

].

FCS_TLSC_EXT.1.6

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

FCS_TLSC_EXT.1.7
The TSF shall prohibit the use of the following extensions:

- Early data extension
- Post-handshake client authentication according to RFC 8446, Section 4.2.6.

FCS_TLSC_EXT.1.8
The TSF shall [not use PSKs].

FCS_TLSC_EXT.1.9
The TSF shall [reject [TLS 1.2] renegotiation attempts].

5.2.2.12 FCS_TLSS_EXT.1 TLS SERVER PROTOCOL

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

- TLS_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

] and no other ciphersuites.

FCS_TLSS_EXT.1.2
The TSF shall authenticate itself using X.509 certificate(s) using [ECDSA over NIST curves [secp256r1] and no other curves].

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

- EC Diffie-Hellman key agreement over NIST curves [secp256r1, secp384r1, secp521r1] and no other curves;

].

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

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

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

- Early data extension

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

FCS_TLSS_EXT.1.8
The TSF shall [reject [TLS 1.2] renegotiation attempts].

5.2.3 IDENTIFICATION AND AUTHENTICATION (FIA)

5.2.3.1 FIA_AFL.1 AUTHENTICATION FAILURE HANDLING

FIA_AFL.1.1
The TSF shall detect when an Administrator configurable positive integer within *[1-999]* unsuccessful authentication attempts occur related to *Administrators attempting to authenticate remotely using a password*.

FIA_AFL.1.2
When the defined number of unsuccessful authentication attempts has been met, the TSF shall [prevent the offending Administrator from successfully establishing a remote session using any authentication method that involves a password until [action to unlock] is taken by an Administrator*]*.

5.2.3.2 FIA_PMG_EXT.1 PASSWORD MANAGEMENT

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

a) Passwords shall be able to be composed of any combination of upper and lower case letters, numbers and the following special characters: [*"!", "@", "#", "$", "%", "^", "&", "*", "(", ")",[" ", "'", "'", "+", ",", "-", ".", "/", ":", ";", "<", "=", ">", "?", "@", "[", "\", "]", "_", "`", "{", "|", "}", "~"*]];

b) Minimum password length shall be *configurable to between [8] and [16] characters.*

5.2.3.3 FIA_UIA_EXT.1 USER IDENTIFICATION AND AUTHENTICATION

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

- Display the warning banner in accordance with FTA_TAB.1;
- [*automated generation of cryptographic keys*].

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

FIA_UIA_EXT.1.3 [TD0900]
The TSF shall provide the following remote authentication mechanisms [Web GUI password] and [external authentication server]. The TSF shall provide the following local authentication mechanisms: [password-based].

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

5.2.3.4 FIA_UAU.7 PROTECTED AUTHENTICATION FEEDBACK (REFINEMENT)

FIA_UAU.7.1
The TSF shall provide only *obscured feedback* to the **administrative** user while the authentication is in progress **at the local console**.

5.2.3.5 FIA_X509_EXT.1/REV X.509 CERTIFICATE VALIDATION

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

- RFC 5280 certificate validation and certification path validation **supporting a minimum path length of three certificates**.
- The certification path must terminate with a trusted CA certificate designated as a trust anchor.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of the certificate using [the Online Certificate Status Protocol (OCSP) as specified in RFC 6960].
- The TSF shall validate the extendedKeyUsage field according to the following rules:
  - *Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.*
  - *Server certificates presented for DTLS/TLS shall have the Server Authentication purpose(id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.*
  - *Client certificates presented for DTLS/TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsagefield.*
  - *OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose(id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.*

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

5.2.3.6 FIA_X509_EXT.2 X.509 CERTIFICATE AUTHENTICATION

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

FIA_X509_EXT.2.2
When the TSF cannot establish a connection to determine the validity of a certificate, the TSF shall [allow the Administrator to choose whether to accept the certificate in these cases].

5.2.3.7 FIA_X509_EXT.3 X.509 CERTIFICATE REQUESTS

FIA_X509_EXT.3.1
The TSF shall generate a Certificate Request as specified by RFC 2986 and be able to provide the following information in the request: public key and [device-specific information, Common Name, Organization, Organizational Unit, Country].

FIA_X509_EXT.3.2
The TSF shall validate the chain of certificates from the Root CA upon receiving the CA Certificate Response.

5.2.4 SECURITY MANAGEMENT (FMT)

5.2.4.1 FMT_MOF.1/MANUALUPDATE MANAGEMENT OF SECURITY FUNCTIONS BEHAVIOR

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

5.2.4.2 FMT_MOF.1/SERVICES MANAGEMENT OF SECURITY FUNCTIONS BEHAVIOR

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

5.2.4.3 FMT_MOF.1/FUNCTIONS MANAGEMENT OF SECURITY FUNCTIONS BEHAVIOUR

FMT_MOF.1.1/Functions
The TSF shall restrict the ability to [determine the behaviour of, modify the behaviour of] *the functions* [transmission of audit data to an external IT entity] to Security Administrators.

5.2.4.4 FMT_MTD.1/COREDATA MANAGEMENT OF TSF DATA

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

5.2.4.5 FMT_MTD.1/CRYPTOKEYS MANAGEMENT OF TSF DATA

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

5.2.4.6 FMT_SMF.1 SPECIFICATION OF MANAGEMENT FUNCTIONS

FMT_SMF.1.1 [TD0880]
The TSF shall be capable of performing the following management functions:

- *Ability to administer the TOE remotely;*
- *Ability to configure the access banner;*

- *Ability to configure the remote session inactivity time before session termination;*
- *Ability to update the TOE, and to verify the updates using digital signature capability prior to installing those updates;*
- [
    - Ability to start and stop services;
    - Ability to modify the behaviour of the transmission of audit data to an external IT entity;
    - 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 manage the cryptographic keys;
    - Ability to configure the cryptographic functionality;
    - 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 generate Certificate Signing Request (CSR) and process CA certificate response;
    - Ability to administer the TOE locally;
    - Ability to configure the local session inactivity time before session termination or locking;
    - Ability to configure the authentication failure parameters for FIA_AFL.1].

5.2.4.7 FMT_SMR.2 RESTRICTIONS ON SECURITY ROLES

FMT_SMR.2.1
The TSF shall maintain the roles:

- *Security Administrator.*

FMT_SMR.2.2
The TSF shall be able to associate users with roles.

FMT_SMR.2.3
The TSF shall ensure that the conditions

- *The Security Administrator role shall be able to administer the TOE remotely*

are satisfied.

5.2.5 PROTECTION OF THE TSF (FPT)

5.2.5.1 FTP_APW_EXT.1 PROTECTION OF ADMINISTRATOR PASSWORDS

FPT_APW_EXT.1.1
The TSF shall store administrative passwords in non-plaintext form.

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

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

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

5.2.5.3 FPT_STM_EXT.1 RELIABLE TIME STAMPS

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

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

5.2.5.4 FPT_TST_EXT.1 TSF TESTING

FPT_TST_EXT.1.1 [TD0836]
The TSF shall run a suite of the following self-tests:

- During initial start-up (on power on) to verify the integrity of the TOE firmware and software;
- Prior to providing any cryptographic service and [at no other time] to verify correct operation of cryptographic implementation necessary to fulfil the TSF;
- [no other] self-tests [*none*].

to demonstrate the correct operation of the TSF.

FPT_TST_EXT.1.2
The TSF shall respond to [all failures] by [*halting or disabling the affected functionality*].

5.2.5.5 FPT_TUD_EXT.1 TRUSTED UPDATE

FPT_TUD_EXT.1.1
The TSF shall provide *Security Administrators* the ability to query the currently executing version of the TOE firmware/software and [no other TOE firmware/software version].

FPT_TUD_EXT.1.2
The TSF shall provide *Security Administrators* the ability to manually initiate updates to TOE firmware/software and [no other update mechanism].

FPT_TUD_EXT.1.3
The TSF shall provide means to authenticate firmware/software updates to the TOE using a [digital signature] prior to installing those updates.

5.2.6 TOE ACCESS (FTA)

5.2.6.1 FTA_SSL_EXT.1 TSF-INITIATED SESSION LOCKING

FTA_SSL_EXT.1.1
The TSF shall, for local interactive sessions, [

- terminate the session]

after a Security Administrator-specified time period of inactivity

5.2.6.2 FTA_SSL.3 TSF-INITIATED TERMINATION

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

5.2.6.3 FTA_SSL.4 USER-INITIATED TERMINATION

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

5.2.6.4 FTA_TAB.1 DEFAULT TOE ACCESS BANNERS

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

5.2.7 TRUSTED PATH/CHANNELS (FTP)

5.2.7.1 FTP_ITC.1 INTER-TSF TRUSTED CHANNEL

FTP_ITC.1.1
The TSF shall **be capable of using [TLS] to** provide a **trusted** communication channel between itself and ~~another trusted IT product~~ **authorized IT entities supporting the following capabilities: audit server, [authentication server]** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from ~~modification or~~ disclosure **and detection of modification of the channel data**.

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

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

5.2.7.2 FTP_TRP.1/ADMIN TRUSTED PATH

FTP_TRP.1.1/Admin
The TSF shall **be capable of using [TLS, HTTPS] to** provide a communication path between itself and **authorized** remote **Administrators** ~~users~~ that is logically distinct from other communication paths and provides assured

identification of its end points and protection of the communicated data from disclosure **and provides detection of modification of the channel data**.

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

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

5.3 TOE SFR DEPENDENCIES RATIONALE FOR SFRS

The PP contain(s) all the requirements claimed in this ST. As such, the dependencies are not applicable since the PP has been approved.

5.4 SECURITY ASSURANCE REQUIREMENTS

The TOE assurance requirements for this ST are taken directly from the PP, which is derived from Common Criteria Version 3.1, Revision 5. The assurance requirements are summarized in Table 11.

Table 11 – Security Assurance Requirements

Assurance Class	Assurance Components	Component Description
Security Target	ASE_CCL.1	Conformance claims
	ASE_ECD.1	Extended components definition
	ASE_INT.1	ST introduction
	ASE_OBJ.1	Security objectives for the operational environment
	ASE_REQ.1	Stated security requirements
	ASE_SPD.1	Security problem definition
	ASE_TSS.1	TOE summary specification
Development	ADV_FSP.1	Basic functional specification
Guidance Documents	AGD_OPE.1	Operational user guidance
	AGD_PRE.1	Preparative procedures
Life Cycle Support	ALC_CMC.1	Labelling of the TOE
	ALC_CMS.1	TOE CM coverage
Tests	ATE_IND.1	Independent testing – conformance
Vulnerability Assessment	AVA_VAN.1	Vulnerability survey

5.5 ASSURANCE MEASURES

The TOE satisfied the identified assurance requirements. This section identifies the Assurance Measures applied by Progress Software Corporation to satisfy the assurance requirements. The following table lists the details.

Table 12 – TOE Security Assurance Measures

SAR Component	How the SAR will be met
ASE_TSS.1.1C Refinement	The TSS is used in conjunction with required supplementary information on Entropy.
ADV_FSP.1	The functional specification describes the external interfaces of the TOE; such as the means for a user to invoke a service and the corresponding response of those services. The description includes the interface(s) that enforces a security functional requirement, the interface(s) that supports the enforcement of a security functional requirement, and the interface(s) that does not enforce any security functional requirements. The interfaces are described in terms of their purpose (general goal of the interface), method of use (how the interface is to be used), parameters (explicit inputs to and outputs from an interface that control the behavior of that interface), parameter descriptions (tells what the parameter is in some meaningful way), and error messages (identifies the condition that generated it, what the message is, and the meaning of any error codes).
AGD_OPE.1	The Administrative Guide provides the descriptions of the processes and procedures of how the administrative users of the TOE can securely administer the TOE using the interfaces that provide the features and functions detailed in the guidance.
AGD_PRE.1	The Installation Guide describes the installation, generation, and startup procedures so that the users of the TOE can put the components of the TOE in the evaluated configuration.
ALC_CMC.1 ALC_CMS.1	The Configuration Management (CM) documents describe how the consumer identifies the evaluated TOE. The CM documents identify the configuration items, how those configuration items are uniquely identified, and the adequacy of the procedures that are used to control and track changes that are made to the TOE. This includes details on what changes are tracked and how potential changes are incorporated.
ATE_IND.1	Vendor will provide the TOE for testing.
AVA_VAN.1	Vendor will provide the TOE for testing. Vendor will provide a document identifying the list of software and hardware components.

6. TOE SUMMARY SPECIFICATION

This chapter identifies and describes how the Security Functional Requirements identified above are met by the TOE.

6.1.1 SECURITY AUDIT (FAU)

6.1.1.1 FAU_GEN.1 AUDIT DATA GENERATION (CPP_ND_V3.0E) AND FAU_GEN.2 USER IDENTITY ASSOCIATION (CPP_ND_V3.0E)

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.

6.1.1.2 FAU_STG_EXT.1 PROTECTED AUDIT EVENT STORAGE (CPP_ND_V3.0E)

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 TOE is a standalone TOE that stores audit data locally.

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 TOE stores persistent audit records in a log file locally, 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 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.

Warning is issued by the TOE before the local storage space for audit data is full.

6.1.1.3 FAU_STG_EXT.3 ACTION IN CASE OF POSSIBLE AUDIT DATA LOSS (CPP_ND_V3.0E)

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

6.1.2 CRYPTOGRAPHIC SUPPORT (FCS)

6.1.2.1 FCS_CKM.1 CRYPTOGRAPHIC KEY GENERATION (CPP_ND_V3.0E)

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 Table 14.

6.1.2.2 FCS_CKM.2 CRYPTOGRAPHIC KEY ESTABLISHMENT (CPP_ND_V3.0E)

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.

6.1.2.3 FCS_CKM.4 CRYPTOGRAPHIC KEY DESTRUCTION (CPP_ND_V3.0E)

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 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.

6.1.2.4 FCS_COP.1/DATAENCRYPTION CRYPTOGRAPHIC OPERATIONS (AES DATA ENCRYPTION/DECRYPTION) (CPP_ND_V3.0E)

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.

The relevant NIST CAVP certificate is listed in Table 14.

6.1.2.5 FCS_COP.1/SIGGEN CRYPTOGRAPHIC OPERATION (SIGNATURE GENERATION AND VERIFICATION) (CPP_ND_V3.0E)

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.

The relevant NIST CAVP certificate is listed in Table 14.

6.1.2.6 FCS_COP.1/HASH CRYPTOGRAPHIC OPERATIONS (HASH ALGORITHM) (CPP_ND_V3.0E)

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

The relevant NIST CAVP certificate is listed in Table 14.

6.1.2.7 FCS_COP.1/KEYEDHASH CRYPTOGRAPHIC OPERATION (KEYED HASH ALGORITHM) (CPP_ND_V3.0E)

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

Table 13 – HMAC Algorithm Description

The relevant NIST CAVP certificate is listed in Table 14.

6.1.2.8 FCS_HTTPS_EXT.1 HTTPS PROTOCOL (CPP_ND_V3.0E)

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.

6.1.2.9 FCS_NTP_EXT.1 NTP PROTOCOL (CPP_ND_V3.0E)

The TSF supports time updates using NTPv4. 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.

6.1.2.10 FCS_RBG_EXT.1 RANDOM BIT GENERATION (CPP_ND_V3.0E)

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 Table 14.

6.1.2.11 FCS_TLSC_EXT.1 TLS CLIENT PROTOCOL (CPP_ND_V3.0E)

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.

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. The TOE does not support the out-of-band provisioning of PSKs.

The TLS channel is terminated if verification fails.

SAN is prioritized over CN. The TOE does not enforce canonical format.

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.

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.

6.1.2.12 FCS_TLSS_EXT.1 TLS SERVER PROTOCOL (CPP_ND_V3.0E)

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 ciphersuites in the TOE.

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.

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).

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.

The TSF does not support session resumption or session tickets.

The TOE does not support the out-of-band provisioning of PSKs.

6.1.3 IDENTIFICATION AND AUTHENTICATION (FIA)

6.1.3.1 FIA_AFL.1 AUTHENTICATION FAILURE HANDLING (CPP_ND_V3.0E)

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 administration of the TSF is always possible, because the TSF never locks the local console.

6.1.3.2 FIA_PMG_EXT.1 PASSWORD MANAGEMENT (CPP_ND_V3.0E)

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.

6.1.3.3 FIA_UIA_EXT.1 USER IDENTIFICATION AND AUTHENTICATION (CPP_ND_V3.0E)

Prior to authentication, the TSF only allows users to display the warning banner and automatically generate keys on a new system.

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 TSF uses a username and password to authenticate administrative users at the local console.

6.1.3.4 FIA_UAU.7 PROTECTED AUTHENTICATION FEEDBACK (REFINEMENT) (CPP_ND_V3.0E)

None.

6.1.3.5 FIA_X509_EXT.1/REV X.509 CERTIFICATE VALIDATION (CPP_ND_V3.0E)

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.

6.1.3.6 FIA_X509_EXT.2 X.509 CERTIFICATE AUTHENTICATION (CPP_ND_V3.0E)

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.

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.

6.1.3.7 FIA_X509_EXT.3 X.509 CERTIFICATE REQUESTS (CPP_ND_V3.0E)

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.

6.1.4 SECURITY MANAGEMENT (FMT)

6.1.4.1 FMT_MOF.1/FUNCTIONS MANAGEMENT OF SECURITY FUNCTIONS BEHAVIOUR (CPP_ND_V3.0E)

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.

6.1.4.2 FMT_MOF.1/MANUALUPDATE MANAGEMENT OF SECURITY FUNCTIONS BEHAVIOR (CPP_ND_V3.0E)

The TSF allows the security administrator to initiate manual updates to the TOE software.

6.1.4.3 FMT_MOF.1/SERVICES MANAGEMENT OF SECURITY FUNCTIONS BEHAVIOUR (CPP_ND_V3.0E)

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.

6.1.4.4 FMT_MTD.1/COREDATA MANAGEMENT DATA (CPP_ND_V3.0E)

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.

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.

6.1.4.5 FMT_MTD.1/CRYPTOKEYS MANAGEMENT OF TSF DATA (CPP_ND_V3.0E)

The TOE provides a trust store to manage and store X.509v3 certificates. 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.

6.1.4.6 FMT_SMF.1 SPECIFICATION OF MANAGEMENT FUNCTIONS (CPP_ND_V3.0E)

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.

6.1.4.7 FMT_SMR.2 RESTRICTIONS ON SECURITY ROLES (CPP_ND_V3.0E)

TSF supports a 'bal' account (superuser). Along with that it also supports admin and non-admin accounts like read-only users. 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.

6.1.5 PROTECTION OF THE TSF (FPT)

6.1.5.1 FPT_APW_EXT.1 PROTECTION OF ADMINISTRATOR PASSWORDS (CPP_ND_V3.0E)

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.

6.1.5.2 FPT_SKP_EXT.1 PROTECTION OF TSF DATA (FOR READING OF ALL SYMMETRIC KEYS) (CPP_ND_V3.0E)

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.

6.1.5.3 FPT_STM_EXT.1 RELIABLE TIME STAMPS (CPP_ND_V3.0E)

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.

6.1.5.4 FPT_TST_EXT.1 TSF TESTING (CPP_ND_V3.0E)

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 hash 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.

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.

These tests collectively ensures the TSF is operating correctly.

6.1.5.5 FPT_TUD_EXT.1 TRUSTED UPDATE (CPP_ND_V3.0E)

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. The TSF also allows the Security Administrator to initiate software updates. 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 digital signature on the update using update public key 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.

6.1.6 TOE ACCESS (FTA)

6.1.6.1 FTA_SSL_EXT.1 TSF-INITIATED SESSION LOCKING (CPP_ND_V3.0E)

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.

6.1.6.2 FTA_SSL.3 TSF-INITIATED TERMINATION (CPP_ND_V3.0E)

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.

6.1.6.3 FTA_SSL.4 USER-INITIATED TERMINATION (CPP_ND_V3.0E)

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.

6.1.6.4 FTA_TAB.1 DEFAULT TOE ACCESS BANNERS (CPP_ND_V3.0E)

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.

6.1.7 TRUSTED PATH/CHANNELS (FTP)

6.1.7.1 FTP_ITC.1 INTER-TSF TRUSTED CHANNEL (CPP_ND_V3.0E)

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.

6.1.7.2 FTP_TRP.1/ADMIN TRUSTED PATH (CPP_ND_V3.0E)

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 protocols listed are consistent with those specified in the requirement.

6.2 CAVP ALGORITHM CERTIFICATE DETAILS

Each of these cryptographic algorithms have been validated as identified in the table below.

Table 14 – CAVP Algorithm Certificate References

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
	CBC *mode* and cryptographic key sizes 128 bits, 256 bits that meet: *AES as specified in ISO*		AES-CBC (Key Length: 128, 256)		

SFR	Algorithm Description	Implementation name	CAVP Alg.	CAVP Cert #	TOE Cards
	18033-3, CBC as specified in ISO 10116.				Virtual LoadMaster
	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

SFR	Algorithm Description	Implementation name	CAVP Alg.	CAVP Cert #	TOE Cards
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
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

SFR	Algorithm Description	Implementation name	CAVP Alg.	CAVP Cert #	TOE Cards
					Virtual LoadMaster

6.3 CRYPTOGRAPHIC KEY DESTRUCTION

The table below describes the key zeroization provided by the TOE and as referenced in FCS_CKM.4.

Table 15 – 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

7. ACRONYM TABLE

Table 16 – Acronyms

Acronym	Definition
AES	Advanced Encryption Standard
CC	Common Criteria
CRL	Certificate Revocation List
DTLS	Datagram Transport Layer Security
EP	Extended Package
GUI	Graphical User Interface
IP	Internet Protocol
NDcPP	Network Device Collaborative Protection Profile
NIAP	Nation Information Assurance Partnership
NTP	Network Time Protocol
OCSP	Online Certificate Status Protocol
PP	Protection Profile
RSA	Rivest, Shamir & Adleman
SFR	Security Functional Requirement
SSH	Secure Shell
ST	Security Target
TOE	Target of Evaluation
TLS	Transport Layer Security
TSS	TOE Summary Specification
WUI	Web User Interface