AlienVault USM for Government v4.12 and RT Login CyberC4:Alert v4.12

Security Target

Version 2.2

October 16, 2015

Prepared For

AlienVault

1875 S. Grant Street, Suite 200
San Mateo, CA, USA 94402

Prepared By

Cygnacom Solutions

Table of Contents

Section **Page**

Figures and Tables

Figures **Page**

Tables **Page**

1 Security Target Introduction

1.1 Security Target Reference

ST Title: *AlienVault USM for Government, Version 4.12 and RT Logic CyberC4:Alert v4.12* Security Target

ST Version: v2.2

ST Author: CygnaCom Solutions Inc.

ST Date: *10/16/2015*

Protection Profile: *U.S. Government Standard Protection Profile for Network Devices, Version 1.1, 08 June 2012*

1.2 TOE Reference

TOE Developer: *AlienVault*

Evaluation Sponsor: *AlienVault*

TOE Identification: *AlienVault USM for Government v4.12 and RT Logic CyberC4:Alert v4.12*

CC Identification: *Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 4, September 2012.*

Table 1: TOE Platforms

Platforms	Version	Device Model
AlienVault USM for Government	v4.12.13	All-In-One

Note: The TOE is also offered as an OEM product through RT Logic, known as the CyberC4: Alert. CyberC4:Alert is an OEM version of USM for Government. The products are identical in terms of hardware, code, functionality. There are no differences between the two. CyberC4:Alert is simply rebranded under RT Logic's product offerings using the same documentation for as USM for Government v4.12.

1.3 TOE Overview

1.3.1 *TOE Product Type*

The Target of Evaluation [TOE] is a Network Device as defined by the protection profile: "*A network device is a device composed of hardware and software that is connected to the network and has an infrastructure role in the overall enterprise".*

1.3.2 *TOE Usage*

The TOE is AlienVault's Unified Security Management (USM) for Government v4.12. The TOE is a network appliance that provides centralized network and compliance monitoring functionality. The TOE offers network administrators with the four essential capabilities in a single platform: asset discovery, behavioral monitoring, vulnerability monitoring, and network security monitoring.

The TOE does not implement a proactive response capability and is purely a monitoring system. The TOE is capable of integrating with external security tools to create a unified monitoring solution, but such external tools are considered part of the operational environment and their use and functionality are outside the scope of this evaluation.

All TOE appliances are shipped ready for immediate access through a remote Web Interface or the local console interface. Some basic features are enabled by default. To ensure secure use, the product must be configured prior to being deployed into a production environment as specified in the user guidance.

1.3.3 *TOE Security Functionality*

- Security Audit
    - Generate audit logs for security-relevant events
    - Supports secure communications to remote syslog servers
- Cryptographic Support
    - Validated cryptographic algorithms
    - Data zeroization
- User Data Protection
    - Residual information clearing
- Identification and Authentication
    - Password and user access policies
- Security Management
    - Local and remote administration
- Protection of the TOE Security Function (TSF)
    - Self-test on power-up
    - Trusted update
- TOE Access
    - Role-based access control
    - Session timeout and lockout
- Trusted Path/Channels
    - Trusted path for remote administrators

1.4 TOE Description

AlienVault USM for Government v4.12 (AlienVault USM) or Target of Evaluation (TOE) is a Security Information and Event Management (SIEM) appliance; a network device that allows monitoring of a distributed enterprise network from a single appliance. AlienVault USM is designed to integrate with a broad range of external applications and network devices that are capable of generating security-relevant events.

The TOE is a hardware appliance built on an Intel-based server platform. The embedded software consists of a hardened Operating System (OS) running modular software, where implemented functionality of the appliance is defined by enabling a specific profile on the device.

The TOE can be subdivided into the following profiles: Sensor, Logger, and Server. By default, the All-In-One installation will enable full functionality and all profiles. In the evaluated configuration, the TOE operates as a standalone, non-distributed appliance. The All-In-One configuration of AlienVault USM is pre-installed on a specific hardware is the evaluated configuration for the TOE.

The TOE is capable of integrating with external IT entities, and that includes other instances of the TOE. It is possible to disable the Server and Logger profiles, and only have the Sensor profile enabled on additional instances of the TOE. These custom configurations are not part of the evaluated configuration; in such cases, these additional instances of the TOE are treated as part of the operational environment.

1.5 TOE Architecture

The underlying architecture of the TOE consists of computer hardware that supports a hardened Linux-based OS that manages the disk, memory, and network resources and provides all necessary support to run the embedded modular software. A dedicated cryptographic module provides cryptographic functionality that implements secure communications and protects critical security parameters.

There is no direct user-space access to the underlying OS, and the TOE does not provide any general-purpose computing capabilities other than the limited subset necessary for its operation. A determined administrator with physical access to the hardware device can always gain access to the OS, but such mode of operation is outside the scope of the evaluation.

The TOE can be subdivided into the following profiles:

- *Sensor*
- *Server*
- *Logger*

Figure 1 outlines the TOE Architecture.

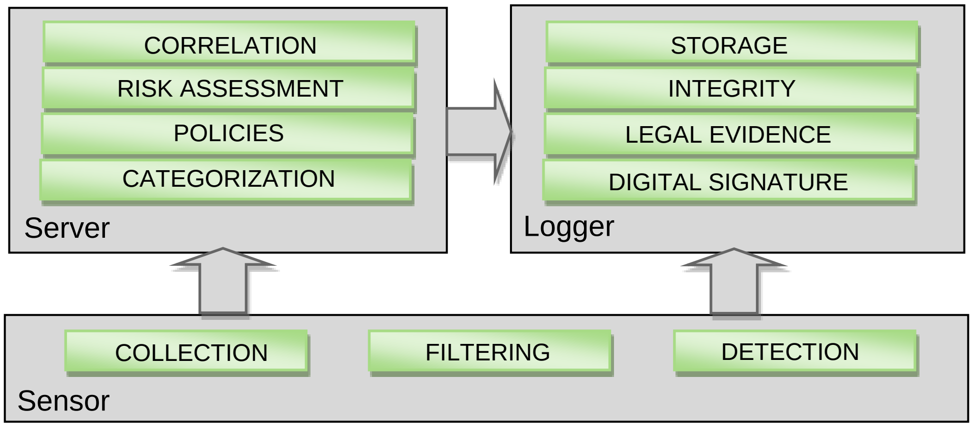

Figure 1: TOE Architecture

1.5.1 *Sensor*

The TOE Sensor has been designed to collect a wide range of information about its local environment, inspect network traffic, detect anomalous activity through various methods, and collect information on suspected attack vectors without affecting overall network performance. The Sensor aggregates all collected information, coordinates threat detection, and monitors compliance within the monitored network. Additional Sensors can be installed on network segments and remote locations, further increasing coverage.

The Sensor integrates an arsenal of monitoring technology into a single logical device, reporting to the network administrators along five different areas:

- Intrusion Detection
- Anomaly Detection
- Vulnerability Detection
- Discovery, Learning and Network Profiling
- Inventory Management

Every TOE deployment, using both active scanning and passive monitoring, creates a highly detailed topography and usage profile of the monitored network. The extensive usage profile is part of the anomaly detection capability. Vulnerability detection algorithms scan and identify latent network threats and can enable network administrator to address them before they can be successfully exploited. The TOE utilizes both signature-based and heuristic methods to detect known and predicted attack vectors in near-real time. The network information gathered by Sensors provides a detailed status in near real-time regarding the network usage of each host in the network monitored by the TOE.

1.5.2 *Server*

Sensors gather and normalize events before sending them to the Server and Logger. This information, stored by the Server, is of vital importance when a host or network attack is in progress. Prior knowledge of existing system settings and vulnerabilities is critical when assessing the risk associated with an attack, prioritizing the severity of the attack, alerting personnel, and implementing appropriate countermeasures.

The Server provides data mining and analytic capabilities, including:

- Risk Assessment
- Event Correlation
- Policy Compliance
- Reporting and Alarms
- Availability Monitoring

The Server component utilizes an internal database to store normalized information, supporting TOE data mining capabilities. The TOE is designed to operate on a busy corporate network and process millions of recorded events per day.

1.5.3 *Logger*

The Logger stores audit events in a forensically secure form. All stored events are digitally signed ensuring their authenticity and integrity.

1.6 TOE Boundaries

1.6.1 *Data*

The data managed by the TOE can be categorized as TSF data and Non-TSF data.

TSF data includes the following:

- Configuration data used to manage and operate the TOE
- Audit data produced by the internal security-relevant events
- Critical security parameters used by cryptographic functions

Non-TSF data includes:

- All data collected from the monitored network
- Analytical data derived from the analyzed events
- Threat and vulnerability signatures

1.6.2 *Physical Boundary*

The TOE’s hardware is based on an Intel Quad-Core Xeon E5 blade server running Debian “Wheezy” 7.8 based on a Linux 3.4 kernel.

The physical boundary of the TOE is the hardware appliance. For the physical boundary, only the *USM All-In-One* hardware configuration included in scope of this evaluation. (See Table 2: Hardware Specifications for details)

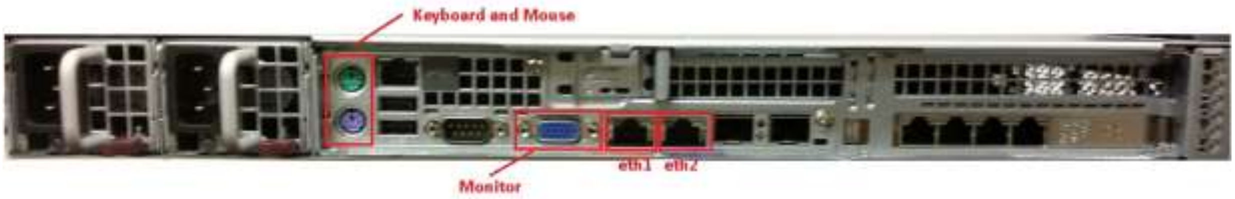

Figure 2: TOE Layout

The TOE has two power supplies that operate in an N+1 configuration. An audible alarm operates if either of the power supplies fail or there is a loss of external power.

Table 2: Hardware Specifications

AlienVault USM All-In-One	
Form Factor	1U
Length x Width x Height (in)	26.6 x 17.2 x1.7
Power Supply	2 x 700/750W
Network Interfaces	6 x 1GbE
CPU	2 x Intel Xeon E5
Storage Capacity (TB)	1.8
Disk Array Configuration	RAID 10
Memory (GB)	24

The TOE can be configured to utilize a number of other components in its operational environment that are *not* included in the evaluation.

- A Management Workstation with a modern browser for Web Interface access.
- A Syslog Server for external storage of the audit log.
- A NTP Server for reliable time.
- A SMTP Server for administrator alert notifications and warnings.
- An external Authentication, Authorization, and Accounting server.
- Additional instances of the TOE.

Figure 3 depicts the TOE boundary with the operational environment and relevant interfaces.

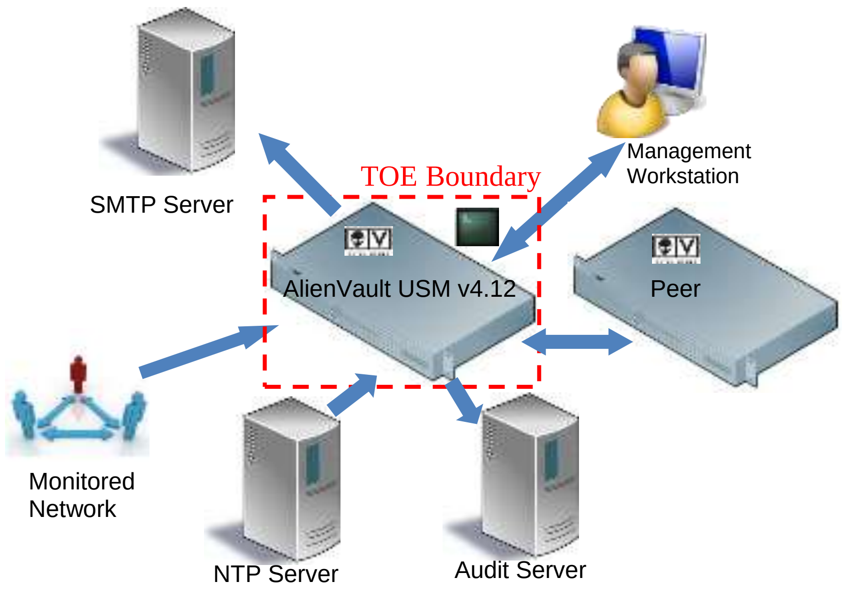

Figure 3: TOE Example Deployment and Boundary

1.6.3 *Logical Boundary*

The logical boundary of the TOE is defined by implemented security functions. These security functions are further described in the following subsections:

1.6.3.1 Security Audit

The TOE generates audit records related to cryptographic functionality, identification and authentication, and management actions. For each security relevant event, the TOE records the date and time, the type of event, the subject identity, and the outcome of the event logged. Auditing is enable by default. The TOE also implements timestamps to ensure that reliable audit information is available. The logs can be accessed through the appropriate menu of the Web Interface. The TOE can be configured to duplicate audit messages to an external Syslog Server.

1.6.3.2 Cryptographic Support

The TOE implements a cryptographic module that performs the following cryptographic operations:

- Secure channel with the following parameters:
  - TLS 1.0 protocol
  - TLS_RSA_WITH_AES_128_CBC_SHA
    TLS_RSA_WITH_AES_256_CBC_SHA
- Random Bit Generation using CTR_DRBG (AES-256)

The TOE uses the cryptographic module to manage session-based plaintext secrets stored in the volatile memory. After the session termination when the secrets are no longer needed, a function call overwrites contents with zeros. The TOE requires a separate, administrator-initiated procedure to clear long-term plaintext secrets from the non-volatile memory.

1.6.3.3 User Data Protection

The TOE implements a residual information clearing mechanism as part of network packet processing. Ingress packets are stored in the managed buffer that allocates dedicated memory space to each packet. Once the packet has been processed, the memory used for that packet is returned back to the pool for reuse. As a result, residual data is never transmitted from the TOE.

1.6.3.4 Identification and Authentication

The TOE uses a Role-Based Access Control (RBAC) structure for restricting system access. Before any other action, each user is identified with a login name and authenticated with a password. Each authorized user is associated with an assigned role and specific permissions that determine their access to TOE features.

User role profiles are defined to provide two axes of permissions:

- Functions that a role can access within the system
- Assets that are accessible for each type of function

The default roles for the TOE are as follows:

- **"Administrator" Role**: this role is used to manage the secure configuration and operation of the TOE
- **"Operator" Role**: this role can view and manipulate data, but cannot access or alter the security functionality of the TOE.

1.6.3.5 Security Management

The TOE allows administrative remote access using a Web Interface. To support remote administration, a management workstation with a web browser that is capable of supporting HTTPS and the TLS protocol must be used. Security management commands are accessible only by authorized administrators with sufficient permissions.

1.6.3.6 Protection of the TSF

The TOE protects against tampering and unauthorized data disclosure by using dedicated communication channels protected by cryptographic means and access control methods. The TOE enforces user access controls and restricts access to the system and data to identified and authorized users. The TOE performs self-tests designed to detect when its core functionality is failing or the TOE has been tampered with. The TOE is designed to periodically check for updates, but updates are applied manually. All updates are protected from tampering by the use of published hash. The TOE also ensures that reliable time information is available for both log accountability and synchronization with the operating environment.

1.6.3.7 TOE access

Access to TOE management functions is restricted to identified, authenticated and authorized users. The TOE implements a message of the day banner displayed during each management session. The TOE enforces a configurable inactivity timer after which a management session is automatically terminated by the TOE. Once a session (local or remote) has been terminated, the TOE requires the user to re-authenticate.

1.6.3.8 Trusted path/channels

The TOE ensures integrity and disclosure protection of remote administrative sessions by implementing HTTPS /TLS. The TOE protects communication with various optional operational environment components, such as a Syslog Server using TLS connection. If the negotiation of a secure session with the TOE fails, a connection will not be established.

1.6.4 *Excluded Functionality*

The TOE supports a number of features that are not part of the core functionality. These features are *not* included in the scope of the evaluation:

- Any integration and/or communication with authentication servers such as Remote Authentication Dial-In User Service (RADIUS) and Terminal Access Controller Access-Control Systems (TACACS) is excluded from the evaluated configuration.
- Remote management using console interface (SSHv2) is excluded and disabled by default.
- Use of the SNMP functionality is excluded and it is disabled by default. The use of SNMPv3 is not restricted; however, it is an excluded functionality in NDPP evaluations.
- Use of the SMTP for sending out automated alerts is outside the scope of NDPP evaluation and as a result is not evaluated.

1.6.5 *TOE Guidance and Reference Documents*

The *AlienVault USM* documentation set includes *a full set of online admin guides.* The following product guidance documents are provided with the TOE*:*

Table 3: User Guidance Documents

Identifier	Edition	Reference Title
AVUG-00001	09	Configuration for Common Criteria
AVUG-00107	01	User Management Guide
AVUG-00116	01	Proxy Configuration
AVUG-00127	01	HIDS Deployment on Windows
AVUG-00131	01	Lifecycle of a Log
AVUG-00133	01	Active Directory Integration
AVUG-00135	01	USM Intrusion Detection
AVUG-00153	01	Send Emails Triggered by Events
AVUG-00160	01	Policy Management Fundamentals
AVUG-00161	01	HIDS File Integrity Monitoring
AVUG-00163	01	Correlation Reference Guide
AVUG-00164	01	Customizing Correlation Directives or Cross Correlation Rules
AVUG-00185	01	Netflow Collection

The most up-to-date versions of the documentation can be accessed on the AlienVault website: www.alienvault.com/documentation

Table 4: ST Reference Documents

Reference Title	ID
Common Criteria for Information Technology Security Evaluation, CCMB-2012-09-002*, Version 3.1, Revision 4*	[CC]
Security Requirements for Network Devices Errata #3, 3 November 2014	[ERRATA]
U.S. Government Standard Protection Profile for Network Devices, Version 1.1, 08 June 2012	[NDPP]

2 Conformance Claims

2.1 Common Criteria Conformance Claim

This Security Target [ST] and the Target of Evaluation [TOE] are conformant to the following Common Criteria [CC] specifications:

- *Common Criteria for Information Technology Security Evaluation Part 2: Security Functional Components,* Version 3.1, Revision 4, September 2012, CCMB-2012-09-002
    - Part 2 Conformant with additional extended functional components as specified by the protection profile.
- *Common Criteria for Information Technology Security Evaluation Part 3: Security Assurance Components,* Version 3.1, Revision 4, September 2012, CCMB-2012-09-003
    - Part 3 Conformant with additional assurance activities as specified by the protection profile.

2.2 Protection Profile Claim

The TOE claims *exact* Compliance to *U.S. Government Standard Protection Profile for Network Devices*, 08 June 2012, Version 1.1 [NDPP] as changed/clarified by *Security Requirements for Network Devices Errata #3*, 3 November 2014 [ERRATA3]

2.3 Package Claim

The TOE does not claim to be conformant with any pre-defined packages.

2.4 Conformance Rationale

This security target claims exact conformance to only one Protection Profile [PP] – the NDPP.

The security problem definition of this ST is consistent with the statement of the security problem definition in the PP, as the ST claims *exact* conformance to the PP and no other threats, organizational security policies, or assumptions are added.

The security objectives of this ST are consistent with the statement of the security objectives in the PP as the ST claims *exact* conformance to the PP and no other security objectives are added.

The security requirements of this ST are consistent with the statement of the security requirements in the PP as the ST claims *exact* conformance to the PP.

3 Security Problem Definition

The security problem to be addressed by the TOE is described by threats and policies. It can be described by the following broad categories:

- Communication with the TOE
- Malicious Updates
- Undetected System Activity
- Accessing the TOE
- User Data Disclosure
- The Security Function Failure

3.1 Threats

This section identifies the threats to the TOE specified by the NDPP, applied verbatim.

Table 5: TOE Threats (formal)

Threat Name	Threat Definition
T.ADMIN_ERROR	An administrator may unintentionally install or configure the TOE incorrectly, resulting in ineffective security mechanisms.
T.TSF_FAILURE	Security mechanisms of the TOE may fail, leading to a compromise of the TSF.
T.UNDETECTED_ACTIONS	Malicious remote users or external IT entities may take actions that adversely affect the security of the TOE. These actions may remain undetected and thus their effects cannot be effectively mitigated.
T.UNAUTHORIZED_ACCESS	A user may gain unauthorized access to the TOE data and TOE executable code. A malicious user, process, or external IT entity may masquerade as an authorized entity in order to gain unauthorized access to data or TOE resources. A malicious user, process, or external IT entity may misrepresent itself as the TOE to obtain identification and authentication data.
T.UNAUTHORIZED_UPDATE	A malicious party attempts to supply the end user with an update to the product that may compromise the security features of the TOE.
T.USER_DATA_REUSE	User data may be inadvertently sent to a destination not intended by the original sender.

3.2 Organizational Security Policies (OSPs)

This section identifies the organizational security policies applicable to the TOE specified by the NDPP, applied verbatim.

Table 6: Organizational Security Policies (formal)

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

3.3 Assumptions

This section identifies assumptions applicable to the TOE specified by the NDPP, applied verbatim.

Table 7: TOE Assumptions (formal)

Assumption Name	Assumption Definition
A.NO_GENERAL_PURPOSE	It is assumed that 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.
A.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the environment.
A.TRUSTED_ADMIN	TOE Administrators are trusted to follow and apply all administrator guidance in a trusted manner.

4 Security Objectives

This section defines the security objectives of the TOE and its supporting environment. Combined, they address threats and implement policies that are imposed by standards or regulation.

The TOE objectives are:

- Protected Communications
- Verifiable Updates
- System Monitoring
- Display Banner
- Administration
- Residual Information Clearing
- Session Locking
- Self-Tests

The operational environment objectives are:

- No general-purpose computing
- Physical security
- Trusted administrators

4.1 Security Objectives for the TOE

This section identifies Security Objectives for the TOE as specified in the NDPP, applied verbatim.

Table 8: TOE Security Objectives (formal)

Objective Name	TOE Security Objective Definition
O.PROTECTED_COMMUNICATIONS	The TOE will provide protected communication channels for administrators, other parts of a distributed TOE, and authorized IT entities.
O.VERIFIABLE_UPDATES	The TOE will provide the capability to help ensure that any updates to the TOE can be verified by the administrator to be unaltered and (optionally) from a trusted source.
O.SYSTEM_MONITORING	The TOE will provide the capability to generate audit data and send those data to an external IT entity.
O.DISPLAY_BANNER	The TOE will display an advisory warning regarding use of the TOE.
O.TOE_ADMINISTRATION	The TOE will provide mechanisms to ensure that only administrators are able to log in and configure the TOE, and provide protections for logged-in administrators.
O.RESIDUAL_INFORMATION_CLEARING	The TOE will ensure that any data contained in a protected resource is not available when the resource is reallocated.

Objective Name	TOE Security Objective Definition
O.SESSION_LOCK	The TOE shall provide mechanisms that mitigate the risk of unattended sessions being hijacked.
O.TSF_SELF_TEST	The TOE will provide the capability to test some subset of its security functionality to ensure it is operating properly.

4.2 Security Objectives for the Operational Environment

This section identifies the Security Objectives for the Operational Environment applicable to the TOE as specified in the NDPP, applied verbatim.

Table 9: Security Objectives for the Operational Environment (formal)

Objective Name	Environmental Security Objective Definition
OE.NO_GENERAL_PURPOSE	There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE.
OE.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.
OE.TRUSTED_ADMIN	TOE Administrators are trusted to follow and apply all administrator guidance in a trusted manner.

5 Extended Components Definition

The components listed in the following table have been defined in *U.S. Government Standard Protection Profile for Network Devices, 08 June 2012, Version 1.1* [NDPP] and clarified by *Security Requirements for Network Devices Errata #3*, 3 November 2014 [ERRATA3].

The extended components are denoted by adding "_EXT" in the component name.

Table 10: Extended Components

Item	SFR ID	SFR Title
1	FAU_STG_EXT.1	Extended: External Audit Trail Storage
2	FCS_CKM_EXT.4	Extended: Cryptographic Key Zeroization
3	FCS_HTTPS_EXT.1	Extended: HTTPS
4	FCS_RBG_EXT.1	Extended: Cryptographic Operation (Random Bit Generation)
5	FCS_TLS_EXT.1	Extended: TLS
6	FIA_PMG_EXT.1	Extended: Password Management
7	FIA_UAU_EXT.2	Extended: Password-based Authentication Mechanism
8	FIA_UIA_EXT.1	Extended: User Identification and Authentication
9	FPT_APW_EXT.1	Extended: Protection of Administrator Passwords
10	FPT_SKP_EXT.1	Extended: Protection of TSF Data (for reading of all symmetric keys)
11	FPT_TST_EXT.1	Extended: TSF Testing
12	FPT_TUD_EXT.1	Extended: Trusted Update
13	FTA_SSL_EXT.1	Extended: TSF-initiated Session Locking

5.1 Extended Security Functional Components Rationale

All extended security functional components are sourced directly from the protection profile or addendum and applied verbatim.

6 Security Requirements

6.1 Security Functional Requirements

Conventions
The following conventions have been applied in this document:

- **Security Functional Requirements –** Part 2 of the CC defines the approved set of operations that may be applied to functional requirements: iteration, assignment, selection, and refinement.
    - **Iteration**: allows a component to be used more than once with varying operations. In the ST, iteration is indicated by a letter in parenthesis placed at the end of the component. For example FDP_ACC.1 (a) and FDP_ACC.1 (b) indicate that the ST includes two iterations of the FDP_ACC.1 requirement, "a" and "b".
    - **Assignment**: allows the specification of an identified parameter. Assignments are indicated using bold italics and are surrounded by brackets (e.g., ***[assignment]).***
    - **Selection**: allows the specification of one or more elements from a list. Selections are indicated using bold italics and are surrounded by brackets (e.g., ***[selection]***).
    - **Refinement**: are identified with "**Refinement:**" right after the short name. Additions to the CC text are specified in ***<u>italicized bold and underlined text</u>***.

Note: Operations already performed in the [NDPP] are not identified in this Security Target

- **Explicitly stated Security Functional Requirements** (i.e., those not found in Part 2 of the CC) are identified "_EXT" in the component name.)

- **Case** - [NDPP] uses an additional convention which defines parts of an SFR that apply only when corresponding selections are made or some other identified conditions exist. Only the applicable cases are identified in this ST.

All SFRs are based on requirements defined in Part 2 of the Common Criteria or defined in the *U.S. Government Standard Protection Profile for Network Devices*, 08 June 2012, Version 1.1 [NDPP] and changed/clarified by *Security Requirements for Network Devices Errata #3*, 3 November 2014 [ERRATA3]

Table 11: TOE Security Functional Components

No.	Component	Component Name
1	FAU_GEN.1	Audit Data Generation
2	FAU_GEN.2	User Identity Association
3	FAU_STG_EXT.1	Extended: External Audit Trail Storage
4	FCS_CKM.1	Cryptographic Key Generation (for asymmetric keys)
5	FCS_CKM_EXT.4	Extended: Cryptographic Key Zeroization
6	FCS_COP.1 (1)	Cryptographic Operation (for data encryption/decryption)
7	FCS_COP.1 (2)	Cryptographic Operation (for cryptographic signature)
8	FCS_COP.1 (3)	Cryptographic Operation (for cryptographic hashing)
9	FCS_COP.1 (4)	Cryptographic Operation (for keyed-hash message authentication)
10	FCS_RBG_EXT.1	Extended: Cryptographic Operation (Random Bit Generation)
11	FCS_HTTPS_EXT.1	Extended: HTTPS
12	FCS_TLS_EXT.1	Extended: TLS
13	FDP_RIP.2	Full Residual Information Protection
14	FIA_PMG_EXT.1	Extended: Password Management
15	FIA_UAU.7	Protected Authentication Feedback
16	FIA_UAU_EXT.2	Extended: Password-based Authentication Mechanism
17	FIA_UIA_EXT.1	Extended: User Identification and Authentication
18	FMT_MTD.1	Management of TSF Data (for general TSF data)
19	FMT_SMF.1	Specification of Management Functions
20	FMT_SMR.2	Restrictions on Security Roles
21	FPT_APW_EXT.1	Extended: Protection of Administrator Passwords
22	FPT_SKP_EXT.1	Extended: Protection of TSF Data (for reading of all symmetric keys)
23	FPT_STM.1	Reliable Time Stamps
24	FPT_TST_EXT.1	Extended: TSF Testing
25	FPT_TUD_EXT.1	Extended: Trusted Update
26	FTA_SSL.3	TSF-initiated Termination
27	FTA_SSL.4	User-initiated Termination
28	FTA_SSL_EXT.1	Extended: TSF-initiated Session Locking
29	FTA_TAB.1	Default TOE Access Banners
30	FTP_ITC.1	Inter-TSF trusted channel
31	FTP_TRP.1	Trusted Path

6.1.1 *Security Audit (FAU)*

6.1.1.1 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) All auditable events for the not specified level of audit; and
c) All administrative actions;
d) *[Specifically defined auditable events listed in **the “Auditable Events” table**]*.

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

a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event; and
b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, *[information specified in column three of **the “Auditable Events” table**.]*

Table 12: Auditable Events

Component	Event	Details
FAU_GEN.1	Start-up and shutdown of the audit functions; All auditable events for the not specified level of audit; and All administrative actions.	No additional information.
FAU_GEN.2	None.	No additional information.
FAU_STG_EXT.1	None.	No additional information.
FCS_CKM.1	None.	No additional information.
FCS_CKM_EXT.4	None.	No additional information.
FCS_COP.1(1)	None.	No additional information.
FCS_COP.1(2)	None.	No additional information.
FCS_COP.1(3)	None.	No additional information.
FCS_COP.1(4)	None.	No additional information.
FCS_RBG_EXT.1	None.	No additional information.
FCS_HTTPS_EXT.1	Failure to establish a HTTPS Session.	Reason for failure.
	Establishment/Termination of a HTTPS session.	Non-TOE endpoint of connection (IP address) for both successes and failures.
FCS_TLS_EXT.1	Failure to establish a TLS Session. Establishment/Termination of a TLS session.	Reason for failure. Non-TOE endpoint of connection (IP address) for both successes and failures.
FDP_RIP.2	None.	No additional information.
FIA_PMG_EXT.1	None.	No additional information.

Component	Event	Details
FIA_UAU.7	None.	No additional information.
FIA_UAU_EXT.2	All use of the authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UIA_EXT.1	All use of the identification and authentication mechanism.	Provided user identity, origin of the attempt (e.g., IP address).
FMT_MTD.1	None.	No additional information.
FMT_SMF.1	None.	No additional information.
FMT_SMR.2	None.	No additional information.
FPT_APW_EXT.1	None.	No additional information.
FPT_SKP_EXT.1	None.	No additional information.
FPT_STM.1	Changes to the time.	The old and new values for the time. Origin of the attempt (e.g., IP address).
FPT_TST_EXT.1	None.	No additional information.
FPT_TUD_EXT.1	Initiation of update.	No additional information.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	No additional information.
FTA_SSL.4	The termination of an interactive session.	No additional information.
FTA_SSL_EXT.1	Any attempts at unlocking of an interactive session.	No additional information.
FTA_TAB.1	None.	No additional information.
FTP_ITC.1	Initiation of the trusted channel. Termination of the trusted channel. Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempt.
FTP_TRP.1	Initiation of the trusted channel. Termination of the trusted channel. Failures of the trusted path functions.	Identification of the claimed user identity.

6.1.1.2 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.

6.1.1.3 FAU_STG_EXT.1 Extended: External Audit Trail Storage

FAU_STG_EXT.1.1 The TSF shall be able to ***[selection: transmit the generated audit data to an external IT entity]*** using a trusted channel implementing the ***[selection: TLS]*** protocol.

6.1.2 *Cryptographic Support (FCS)*

6.1.2.1 FCS_CKM.1 Cryptographic Key Generation (for asymmetric keys)

FCS_CKM.1.1 The TSF shall generate asymmetric cryptographic keys used for key establishment in accordance with

[selection:

- ***NIST Special Publication 800-56B, "Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography" for RSA-based key establishment schemes***

]

and specified cryptographic key sizes equivalent to, or greater than, a symmetric key strength of 112 bits.

6.1.2.2 FCS_CKM_EXT.4 Extended: Cryptographic Key Zeroization

FCS_CKM_EXT.4.1 The TSF shall zeroize all plaintext secret and private cryptographic keys and CSPs when no longer required.

6.1.2.3 FCS_COP.1(1) Cryptographic Operation (for data encryption/decryption)

FCS_COP.1.1(1) The TSF shall perform [encryption and decryption] in accordance with a specified cryptographic algorithm [AES operating in ***[selection: CBC mode]]*** and cryptographic key sizes 128-bits, and 256-bits that meets the following:

- **FIPS PUB 197, "Advanced Encryption Standard (AES)"**
- ***[Selection: NIST SP 800-38A]***

6.1.2.4 FCS_COP.1(2) Cryptographic Operation (for cryptographic signature)

FCS_COP.1.1(2) The TSF shall perform [cryptographic signature services] in accordance with a ***[selection:***

(1) RSA Digital Signature Algorithm (rDSA) with a key size (modulus) of 2048 bits or greater,]

that meets the following**:**

Case: **RSA Digital Signature Algorithm**

- **FIPS PUB 186-3, *"Digital Signature Standard"***

6.1.2.5 FCS_COP.1(3) Cryptographic Operation (for cryptographic hashing)

FCS_COP.1.1(3) The TSF shall perform [cryptographic hashing services] in accordance with a specified cryptographic algorithm ***[selection: SHA-1, SHA-256, SHA-512]*** and message digest sizes ***[selection: 160, 256, 512]*** bits that meet the following: FIPS Pub 180-3, “Secure Hash Standard.”

6.1.2.6 FCS_COP.1(4) Cryptographic Operation (for keyed-hash message authentication)

FCS_COP.1.1(4) The TSF shall perform [keyed-hash message authentication] in accordance with a specified cryptographic algorithm HMAC***-[selection: SHA-1, SHA-256, SHA-512]***, key size ***[assignment: 160, 256, 512 bits]***, and message digest sizes ***[selection: 160, 256, 512]*** bits that meet the following: FIPS Pub 198-1, "*The Keyed-Hash Message Authentication Code”*, and FIPS Pub 180-3, *“Secure Hash Standard.”*

6.1.2.7 FCS_RBG_EXT.1 Extended: Cryptographic Operation (Random Bit Generation)

FCS_RBG_EXT.1.1 The TSF shall perform all random bit generation (RBG) services in accordance with ***[selection: NIST Special Publication 800-90 using [CTR_DRBG (AES)]]*** seeded by an entropy source that accumulated entropy from ***[a software-based noise source].***

FCS_RBG_EXT.1.2 The deterministic RBG shall be seeded with a minimum of ***[selection: 256 bits]*** of entropy at least equal to the greatest security strength of the keys and hashes that it will generate.

6.1.2.8 FCS_HTTPS_EXT.1 Extended: HTTPS

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

FCS_HTTPS_EXT.1.2 The TSF shall implement HTTPS using TLS as specified in FCS_TLS_EXT.

6.1.2.9 FCS_TLS_EXT.1 Extended: TLS

FCS_TLS_EXT.1.1 The TSF shall implement one or more of the following protocols [TLS 1.0 (RFC 2246)] supporting the following cipher suites:

Mandatory Cipher suites:
TLS_RSA_WITH_AES_128_CBC_SHA

Optional Cipher suites:
[selection:
TLS_RSA_WITH_AES_256_CBC_SHA*].*

6.1.3 *User Data Protection (FDP)*

6.1.3.1 FDP_RIP.2 Full Residual Information Protection

FDP_RIP.2.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the ***[selection: allocation of the resource to]*** all objects.

6.1.4 *Identification and Authentication (FIA)*

6.1.4.1 FIA_PMG_EXT.1 Extended: Password Management

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

1. Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters: ***[selection: “!”, “@”, “#”, “$”, “%”, “&”, “(“, “)”, “***space”,”.”, “_”, “-“, “=”, “+”, “:”, “;”, “,” “[“, “]”, “{“, “}”, “|” , “¡” ,“¿” ,“º”, “ª”, ***“•”];***

2. Minimum password length shall settable by the Security Administrator, and support passwords of 15 characters or greater;

6.1.4.2 FIA_UIA_EXT.1 Extended: 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;
- ***[selection: no other actions]***

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

6.1.4.3 FIA_UAU_EXT.2 Extended: Password-based Authentication Mechanism

FIA_UAU_EXT.2.1 The TSF shall provide a local password-based authentication mechanism, ***[selection: none]*** to perform administrative user authentication.

6.1.4.4 FIA_UAU.7 Protected Authentication Feedback

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.

6.1.5 *Security Management (FMT)*

6.1.5.1 FMT_MTD.1 Management of TSF Data (for general TSF data)

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

6.1.5.2 FMT_SMF.1 Specification of Management Functions

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

- Ability to administer the TOE locally and remotely;
- Ability to update the TOE, and to verify the updates using ***[selection: published hash]*** capability prior to installing those updates;
- ***[selection:***
  - ***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 cryptographic functionality.]***

6.1.5.3 FMT_SMR.2 Restrictions on Security Roles

FMT_SMR.2.1 The TSF shall maintain the roles:

- Authorized 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:

- Authorized Administrator role shall be able to administer the TOE locally;
- Authorized Administrator role shall be able to administer the TOE remotely;

are satisfied.

6.1.6 *Protection of the TSF (FPT)*

6.1.6.1 FPT_SKP_EXT.1 Extended: 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.

6.1.6.2 FPT_APW_EXT.1 Extended: Protection of Administrator Passwords

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

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

6.1.6.3 FPT_STM.1 Reliable Time Stamps

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

6.1.6.4 FPT_TUD_EXT.1 Extended: Trusted Update

FPT_TUD_EXT.1.1 The TSF provides security administrators the ability to query the current version of the TOE firmware/software.

FPT_TUD_EXT.1.2 The TSF provides security administrators the ability to initiate updates to TOE firmware/software.

FPT_TUD_EXT.1.3 The TSF shall provide a means to verify firmware/software updates to the TOE using a ***[selection: published hash]*** prior to installing those updates.

6.1.6.5 FPT_TST_EXT.1: TSF Testing

FPT_TST_EXT.1.1 The TSF shall run a suite of self-tests during initial start-up (on power on) to demonstrate the correct operation of the TSF.

6.1.7 *TOE Access (FTA)*

6.1.7.1 FTA_SSL_EXT.1 TSF-initiated Session Locking

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

6.1.7.2 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]*.

6.1.7.3 FTA_SSL.4 User-initiated Termination

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

6.1.7.4 FTA_TAB.1 Default TOE Access Banners

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

6.1.8 *Trusted Path/Channels (FTP)*

6.1.8.1 FTP_ITC.1 Inter-TSF Trusted Channel

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

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

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for ***[assignment: transmitting audit records to an audit server].***

6.1.8.2 FTP_TRP.1 Trusted Path

FTP_TRP.1.1 The TSF shall use ***[selection: TLS/HTTPS]*** to provide a trusted communication path between itself and remote administrators that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from disclosure and detection of modification of the communicated data.

FTP_TRP.1.2 The TSF shall permit remote administrators to initiate communication via the trusted path.

FTP_TRP.1.3 TSF shall require the use of the trusted path for initial administrator authentication and all remote administration actions.

6.2 Security Assurance Requirements

6.2.1 *Security Assurance Requirements for the TOE*

This section defines the assurance requirements for the TOE. The assurance activities to be performed by the evaluator are defined in Section 4 and Appendix C of the U.S. Government Standard Protection Profile for Network Devices, [NDPP] and as modified by [ERRATA3]. The TOE security assurance requirements, summarized in the table below, identify the management and evaluative activities required to address the threats identified in [NDPP].

Table 13: NDPP Assurance Components

Assurance Class	Assurance Components	
Development	ADV_FSP.1	Basic Functional Specification
Guidance documents	AGD_OPE.1	Operational User guidance
	AGD_PRE.1	Preparative User guidance
Life cycle support	ALC_CMC.1	Labeling of the TOE
	ALC_CMS.1	TOE CM coverage
Tests	ATE_IND.1	Independent testing - conformance
Vulnerability assessment	AVA_VAN.1	Vulnerability analysis

The following tables state the developer action elements, content and presentation elements and evaluator action elements for each of the assurance components.

Table 14: ADV_FSP.1 Basic Functional Specification

Developer action elements	
ADV_FSP.1.1D	The developer shall provide a functional specification.
ADV_FSP.1.2D	The developer shall provide a tracing from the functional specification to the SFRs.
Content and presentation elements	
ADV_FSP.1.1C	The functional specification shall describe the purpose and method of use for each SFR-enforcing and SFR-supporting TSFI.
ADV_FSP.1.2C	The functional specification shall identify all parameters associated with each SFR-enforcing and SFR-supporting TSFI.
ADV_FSP.1.3C	The functional specification shall provide rationale for the implicit categorization of interfaces as SFR-non-interfering.
ADV_FSP.1.4C	The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification.
Evaluator action elements	
ADV_ FSP.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ADV_ FSP.1.2E	The evaluator shall determine that the functional specification is an accurate and complete instantiation of the SFRs.

Table 15: AGD_OPE.1 Operational User Guidance

Developer action elements	
AGD_OPE.1.1D	The developer shall provide operational user guidance.
Content and presentation elements	
AGD_OPE.1.1C	The operational user guidance shall describe, for each user role, the user-accessible functions and privileges that should be controlled in a secure

	processing environment, including appropriate warnings.
AGD_OPE.1.2C	The operational user guidance shall describe, for each user role, how to use the available interfaces provided by the TOE in a secure manner.
AGD_OPE.1.3C	The operational user guidance shall describe, for each user role, the available functions and interfaces, in particular all security parameters under the control of the user, indicating secure values as appropriate.
AGD_OPE.1.4C	The operational user guidance shall, for each user role, clearly present each type of security-relevant event relative to the user-accessible functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.
AGD_OPE.1.5C	The operational user guidance shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences, and implications for maintaining secure operation.
AGD_OPE.1.6C	The operational user guidance shall, for each user role, describe the security measures to be followed in order to fulfill the security objectives for the operational environment as described in the ST.
AGD_OPE.1.7C	The operational user guidance shall be clear and reasonable.
Evaluator action elements	
AGD_OPE.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

Table 16: AGD_PRE.1 Preparative Procedures

Developer action elements	
AGD_PRE.1.1D	The developer shall provide the TOE, including its preparative procedures.
Content and presentation elements	
AGD_ PRE.1.1C	The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.
AGD_ PRE.1.2C	The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the operational environment in accordance with the security objectives for the operational environment as described in the ST.
Evaluator action elements	
AGD_ PRE.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
AGD_ PRE.1.2E	The evaluator shall apply the preparative procedures to confirm that the TOE can be prepared securely for operation.

Table 17: ALC_CMC.1 Labeling of the TOE

Developer action elements	
ALC_CMC.1.1D	The developer shall provide the TOE and a reference for the TOE.
Content and presentation elements	
ALC_CMC.1.1C	The TOE shall be labeled with its unique reference.
Evaluator action elements	
ALC_CMC.2.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

Table 18: ALC_CMS.1 TOE CM Coverage

Developer action elements	
ALC_CMS.2.1D	The developer shall provide a configuration list for the TOE.
Content and presentation elements	

ALC_CMS.2.1C	The configuration list shall include the following: the TOE itself; and the evaluation evidence required by the SARs.
ALC_CMS.2.2C	The configuration list shall uniquely identify the configuration items.
Evaluator action elements	
ALC_CMS.2.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

Table 19: ATE_IND.1 Independent Testing – Conformance

Developer action elements	
ATE_IND.1.1D	The developer shall provide the TOE for testing.
Content and presentation elements	
ATE_IND.1.1C	The TOE shall be suitable for testing.
Evaluator action elements	
ATE_IND.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ATE_IND.1.2E	The evaluator shall test a subset of the TSF to confirm that the TSF operates as specified.

Table 20: AVA_VAN.1 Vulnerability Survey

Developer action elements	
AVA_VAN.1.1D	The developer shall provide the TOE for testing.
Content and presentation elements	
AVA_VAN.1.1C	The TOE shall be suitable for testing.
Evaluator action elements	
AVA_VAN.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
AVA_VAN.1.2E	The evaluator shall perform a search of public domain sources to identify potential vulnerabilities in the TOE.
AVA_VAN.1.3E	The evaluator shall conduct penetration testing, based on the identified potential vulnerabilities, to determine that the TOE is resistant to attacks performed by an attacker possessing Basic attack potential.

6.2.2 *Security Assurance Requirements Rationale*

This ST conforms to the [NDPP], which draws from the CC Security Assurance Requirements (SARs) to frame the extent to which the evaluator assesses the documentation applicable for the evaluation and performs independent testing.

6.2.3 *Extended Assurance Activities*

The following subsections define the explicit assurance activities presented in the [NDPP] and [ERRATA3] for applicable SAR families. These assurance activities serve to refine the standard SARs previously stated with specific activities to be performed by the evaluators during the course of their evaluation.

6.2.3.1 Class ADV Assurance Activities

Introduction

The Functional Specification (FSP) describes the TOE Security Functions Interfaces (TSFIs). It is not necessary to have a formal or complete specification of these interfaces. Additionally, because TOEs conforming to this PP will necessarily have interfaces to the Operational Environment that are not directly invokable by TOE users, there is little point specifying that such interfaces be described in and of themselves since only indirect testing of such interfaces may be possible. For this PP, the activities for this family should focus on understanding the interfaces presented in the TSS in response to the functional requirements and the interfaces presented in the AGD documentation. No additional FSP documentation is necessary to satisfy the assurance activities specified.

The interfaces that need to be evaluated are characterized through the information needed to perform the assurance activities listed, rather than as an independent, abstract list.

ADV_FSP.1 Activities

There are no specific assurance activities associated with these SARs. The FSP documentation is provided to support the evaluation activities described in [NDPP] Section 4.2, and other activities described for AGD, ATE, and AVA SARs. The requirements on the content of the FSP information is implicitly assessed by virtue of the other assurance activities being performed; if the evaluator is unable to perform an activity because the there is insufficient interface information, then an adequate FSP has not been provided.

6.2.3.2 Class AGD Assurance Activities

Introduction

The guidance documents will be provided with the developer's ST. The guidance must include a description of how the authorized user verifies that the Operational Environment can fulfill its role for the security functionality. The documentation should be in an informal style and readable by an authorized user.

Guidance must be provided for every Operational Environment that the product supports as claimed in the ST. This guidance includes

- instructions to successfully install the TOE in that environment; and
- instructions to manage the security of the TOE as a product and as a component of the larger Operational Environment.

Guidance pertaining to particular security functionality is also provided; specific requirements on such guidance are contained in the assurance activities specified in [NDPP] Section 4.2.

AGD_OPE.1 Activities

Some of the contents of the operational guidance will be verified by the assurance activities in [NDPP] Section 4.2 and evaluation of the TOE according to the CEM. The following additional information is also required.

The operational guidance shall at a minimum list the processes running (or that could run) on the TOE in its evaluated configuration during its operation that are capable of processing data received on the network interfaces (there are likely more than one of these, and this is not limited to the process that "listens" on the network interface). It is acceptable to list all processes running (or that could run) on the TOE in its evaluated configuration instead of attempting to determine just those that process the network data. For each process listed, the administrative guidance will contain a short (e.g., one- or two-line) description of the process' function, and the privilege with which the service runs. "Privilege" includes the hardware privilege level (e.g., ring 0, ring 1), any software privileges specifically associated with the process, and the privileges associated with the user role the process runs as or under.

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

The documentation must describe the process for verifying updates to the TOE, either by checking the hash or by verifying a digital signature. The evaluator shall verify that this process includes the following steps:

1. For hashes, a description of where the hash for a given update can be obtained. For digital signatures, instructions for obtaining the certificate that will be used by the FCS_COP.1 (2) mechanism to ensure that a signed update has been received from the certificate owner. This may be supplied with the product initially, or may be obtained by some other means.

2. Instructions for obtaining the update itself. This should include instructions for making the update accessible to the TOE (e.g., placement in a specific directory).

3. Instructions for initiating the update process, as well as discerning whether the process was successful or unsuccessful. This includes generation of the hash/digital signature.

The TOE will likely contain security functionality that does not fall in the scope of evaluation under this PP. The operational guidance shall make it clear to an administrator which security functionality is covered by the evaluation activities.

AGD_PRE.1 Activities

As indicated in the introduction above, there are significant expectations with respect to the documentation—especially when configuring the operational environment to support TOE functional requirements. The evaluator shall check to ensure that the guidance provided for the TOE adequately addresses all platforms claimed for the TOE in the ST.

6.2.3.3 Class ALC Assurance Activities

Introduction

At the assurance level provided for TOEs conformant to this PP, life-cycle support is limited to end-user-visible aspects of the life-cycle, rather than an examination of the TOE vendor’s development and configuration management process. This is not meant to diminish the critical role that a developer’s practices play in contributing to the overall trustworthiness of a product;

rather, it is a reflection on the information to be made available for evaluation at this assurance level.

ALC_CMC.1 Activities

The evaluator shall check the ST to ensure that it contains an identifier (such as a product name/version number) that specifically identifies the version that meets the requirements of the ST. The evaluator shall ensure that this identifier is sufficient for an acquisition entity to use in procuring the TOE (including the appropriate administrative guidance) as specified in the ST. Further, the evaluator shall check the AGD guidance and TOE samples received for testing to ensure that the version number is consistent with that in the ST. If the vendor maintains a web site advertising the TOE, the evaluator shall examine the information on the web site to ensure that the information in the ST is sufficient to distinguish the product.

ALC_CMS.1 Activities

The “evaluation evidence required by the SARs” in this PP is limited to the information in the ST coupled with the guidance provided to administrators and users under the AGD requirements. By ensuring that the TOE is specifically identified and that this identification is consistent in the ST and in the AGD guidance (as done in the assurance activity for ALC_CMC.1), the evaluator implicitly confirms the information required by this component.

6.2.3.4 Class ATE Assurance Activities

Introduction

Testing is specified for functional aspects of the system as well as aspects that take advantage of design or implementation weaknesses. The former is done through the ATE_IND family, while the latter is through the AVA_VAN family. At the assurance level specified in this PP, testing is based on advertised functionality and interfaces with dependency on the availability of design information. One of the primary outputs of the evaluation process is the test report as specified in the following requirements.

ATE_IND.1 Activities

The evaluator shall prepare a test plan and report documenting the testing aspects of the system. The test plan covers all of the testing actions contained in the CEM and the body of this PP’s Assurance Activities. While it is not necessary to have one test case per test listed in an Assurance Activity, the evaluator must document in the test plan that each applicable testing requirement in the ST is covered.

The test plan identifies the platforms to be tested, and for those platforms not included in the test plan but included in the ST, the test plan provides a justification for not testing the platforms. This justification must address the differences between the tested platforms and the untested platforms, and make an argument that the differences do not affect the testing to be performed. It is not sufficient to merely assert that the differences have no affect; rationale must be provided. If all platforms claimed in the ST are tested, then no rationale is necessary.

The test plan describes the composition of each platform to be tested, and any setup that is necessary beyond what is contained in the AGD documentation. It should be noted that the evaluator is expected to follow the AGD documentation for installation and setup of each platform either as part of a test or as a standard pre-test condition. This may include special test drivers or tools. For each driver or tool, an argument (not just an assertion) should be provided that the driver or tool will not adversely affect the performance of the functionality by the TOE and its platform. This also includes the configuration of the cryptographic engine to be used. The cryptographic algorithms implemented by this engine are those specified by this PP and used by the cryptographic protocols being evaluated.

The test plan identifies high-level test objectives as well as the test procedures to be followed to achieve those objectives. These procedures include expected results. The test report (which could just be an annotated version of the test plan) details the activities that took place when the test procedures were executed, and includes the actual results of the tests. This shall be a cumulative account, so if there was a test run that resulted in a failure; a fix installed; and then a successful re-run of the test, the report would show a “fail” and “pass” result (and the supporting details), and not just the “pass” result.

6.2.3.5 Class AVA Assurance Activities

Introduction

The evaluation lab is expected to survey open sources to discover what vulnerabilities have been discovered in these types of products. In most cases, these vulnerabilities will require sophistication beyond that of a basic attacker. Until penetration tools are created and uniformly distributed to the evaluation labs, the evaluator will not be expected to test for these vulnerabilities in the TOE. The labs will be expected to comment on the likelihood of these vulnerabilities given the documentation provided by the vendor. This information will be used in the development of penetration testing tools and for the development of future PPs.

AVA_VAN.1 Activities

As with ATE_IND, the evaluator shall generate a report to document their findings with respect to this requirement. This report could physically be part of the overall test report mentioned in ATE_IND, or a separate document. The evaluator performs a search of public information to determine the vulnerabilities that have been found in network infrastructure devices and the implemented communication protocols in general, as well as those that pertain to the particular TOE. The evaluator documents the sources consulted and the vulnerabilities found in the report. For each vulnerability found, the evaluator either provides a rationale with respect to its non-applicability, or the evaluator formulates a test (using the guidelines provided in ATE_IND) to confirm the vulnerability, if suitable. Suitability is determined by assessing the attack vector needed to take advantage of the vulnerability. For example, if the vulnerability can be detected by pressing a key combination on boot-up, a test would be suitable at the assurance level of this PP. If exploiting the vulnerability requires expert skills and an electron microscope, for instance, then a test would not be suitable and an appropriate justification would be formulated.

6.2.4 *Extended Assurance Activities*

The extended assurance activities define the explicit activities specified in the [NDPP] and [ERRATA3] for applicable SFR and SAR elements. These activities are detailed in the Assurance Activity Report [AAR].

6.3 Rationale

This ST claims Exact Compliance to *U.S. Government Standard Protection Profile for Network Devices*, 08 June 2012, Version 1.1 [NDPP] as changed/clarified by *Security Requirements for Network Devices Errata #3*, 3 November 2014 [ERRATA3]. Therefore:

- All secure usage assumptions, organizational security policies, and threats are completely covered by security objectives.
- Each objective counters or addresses at least one assumption, organizational security policy, or threat.
- The set of components (requirements) in the ST internally consistent and complete.

6.3.1 *TOE SFR Dependencies*

The following table provides SFR dependency mapping. All SFRs were drawn from the NDPP. For extended components that were derived from SFRs from CC Part 2 dependencies were based on unmodified SFRs. For extended components without baseline equivalent, no dependency was specified.

Table 21: SFR Dependencies

SFR	Dependency	Satisfied by
FAU_GEN.1	FPT_STM.1	FPT_STM.1
FAU_GEN.2	FAU_GEN.1 FAU_UID.1	FAU_GEN.1 FIA_UIA_EXT.1
FAU_STG_EXT.1	FAU_GEN.1	FAU_GEN.1
FCS_CKM.1	FCS_COP.1 FCS_CKM.4	FCS_COP.1(2) FCS_CKM._EXT.4
FCS_CKM_EXT.4	FCS_CKM.1 FCS_CKM.4	FCS_CKM.1 FCS_CKM_EXT.4
FCS_COP.1(1)	FCS_CKM.1 FCS_CKM.4	FCS_CKM.1 FCS_CKM_EXT.4
FCS_COP.1(2)	FCS_CKM.1 FCS_CKM.4	FCS_CKM.1 FCS_CKM_EXT.4
FCS_COP.1(3)	FCS_CKM.1 FCS_CKM.4	FCS_CKM.1 FCS_CKM_EXT.4
FCS_COP.1(4)	FCS_CKM.1 FCS_CKM.4	FCS_CKM.1 FCS_CKM_EXT.4
FCS_RBG_EXT.1	n/a	
FCS_HTTPS_EXT.1	n/a	
FCS_TLS_EXT.1	n/a	
FDP_RIP.2	none	
FIA_PMG_EXT.1	n/a	
FIA_UAU.7	FIA_UAU.1	FIA_UIA_EXT.1

FIA_UAU_EXT.2	FIA_UID.1	FIA_UIA_EXT.1
FIA_UIA_EXT.1	n/a	
FMT_MTD.1	FMT_SMF.1 FMT_SMR.1	FMT_SMF.1 FMT_SMR.1
FMT_SMF.1	none	
FMT_SMR.2	FIA_UID.1	FIA_UIA_EXT.1
FPT_APW_EXT.1	n/a	
FPT_STM.1	none	
FPT_SKP_EXT.1	n/a	
FPT_TST_EXT.1	none	
FPT_TUD_EXT.1	n/a	
FTA_SSL.3	none	
FTA_SSL.4	none	
FTA_SSL_EXT.1	FIA_UAU.1	FIA_UIA_EXT.1
FTA_TAB.1	none	
FTP_ITC.1	none	
FTP_TRP.1	none	

7 TOE Summary Specification

This chapter describes the security functions and provides a high-level description of each SFR. The intended purpose of this description is to enable a general understanding of how the TOE is implemented. The descriptions are intentionally not detailed, and are not a suitable substitution for a detailed specifications found in the guidance documentation.

The TOE consists of the following Security Functions:

- Security Audit
- Cryptographic Support
- User Data Protection
- Identification and Authentication
- Security Management
- Protection of the security functionality
- TOE Access
- Trusted Path/Channels

These Security Functions are implemented by following Security Functional Requirements that in turn map to Security Objectives. See Table 22 for details.

Table 22: Security Functions mapped to Security Objectives

Security Functions	SFR	Security Objectives
Security Audit	FAU_GEN.1	System Monitoring
	FAU_GEN.2	System Monitoring
	FAU_STG_EXT.1	System Monitoring
Cryptographic Support	FCS_CKM.1	Protected Communications
	FCS_CKM_EXT.4	Protected Communications
	FCS_COP.1(1)	Protected Communications
	FCS_COP.1(2)	Protected Communications Verifiable Updates
	FCS_COP.1(3)	Protected Communications Verifiable Updates
	FCS_COP.1(4)	Protected Communications
	FCS_RBG_EXT.1	Protected Communications
	FCS_HTTPS_EXT.1	Protected Communications Administration
	FCS_TLS_EXT.1	Protected Communications
User Data Protection	FDP_RIP.2	Residual Information Clearing
Identification and Authentication	FIA_PMG_EXT.1	Administration
	FIA_UAU.7	Administration
	FIA_UAU_EXT.2	Administration
	FIA_UIA_EXT.1	Administration
Security Management	FMT_MTD.1	Administration
	FMT_SMF.1	Administration
	FMT_SMR.2	Administration
Protection of the security functionality	FPT_APW_EXT.1	Administration
	FPT_SKP_EXT.1	Protected Communications
	FPT_STM.1	System Monitoring
	FPT_TST_EXT.1	Self-Testing

Security Functions	SFR	Security Objectives
	FPT_TUD_EXT.1	Verifiable Updates
TOE Access	FTA_SSL.3	Session Locking
	FTA_SSL.4	Administration
	FTA_SSL_EXT.1	Session Locking
	FTA_TAB.1	Display Banner
Trusted Path/Channels	FTP_ITC.1	Protected Communications
	FTP_TRP.1	Protected Communications

7.1 Security Audit

FAU_GEN.1, FAU_GEN2, FAU_STG_EXT.1, FPT_STM.1

The TOE system processes generate audit records for a wide range of security relevant and operational events. For each security relevant event, the TOE records the date and time, the type of event, the subject identity, and the outcome of the event logged. Events that generate an audit record include a number of system events, as well as all of the events identified in Table 12: Auditable Events.

The TOE implements an internal log storage, where each audit log entry is recorded in an internal log file. Local audit logs are secured against unauthorized access with role-based access control. The logs can be accessed by an authenticated authorized administrator through the appropriate menu of the management interface.

Auditing is enabled and configured to store locally by default. If audit logs are reaching configured capacity, the TOE can be setup through a policy to warn the administrator. In the unlikely event that the audit log file reaches 1.8Tb capacity, the oldest logs are deleted to make room for new entries.

The TOE can also be configured to upload audit logs to an external Syslog Server. A trusted channel is created when the TOE establishes a secure TLS v1.0 session between itself and the external Syslog Server. After the secure session is established, the TOE is designed to forward all audit messages to the listening port created by the secure connection. This ensures that all audit traffic is encapsulated and protected against data disclosure and modification.

The TOE also implements reliable timestamps to ensure accurate audit information is produced. The time is maintained by a hardware clock within the TOE, that can be configured to automatically synchronize with an external time server.

7.2 Cryptographic Support

FCS_CKM.1, FCS_CKM_EXT.4, FCS_COP.1 (1-4), FCS_RBG_EXT.1, FCS_TLS_EXT.1, FCS_HTTPS_EXT.1

The TOE performs all cryptographic operations using a dedicated OpenSSL v2.0.9 cryptographic module that implements all cryptographic security functionality listed in the Table 23. The evaluated configuration requires that the TOE must be configured to use FIPS mode.

Table 23: TOE Cryptography

Requirement Class	Requirement Component	TOE Implementation	Certificate #
FCS: Cryptographic Support	FCS_CKM.1 Cryptographic Key Generation	Implemented by the cryptographic module operating in FIPS mode. The TOE generates all asymmetric cryptographic keys used for key establishment in accordance with NIST SP 800-56B.	**RSA #1834**
	FCS_COP.1(1) Cryptographic Operation (encryption/decryption)	AES-CBC-128 and AES-CBC-256 for data encryption/decryption are implemented to meet FIPS PUB 197, “Advanced Encryption Standard (AES)” in compliance with NIST SP 800-38A. Encryption/decryption is performed by the cryptographic module operating in FIPS mode.	**AES #3566**
	FCS_COP.1(2) Cryptographic Operation (cryptographic signature)	RSA Digital Signature Algorithm (rDSA) with a key size (modulus) of 2048 bits or greater in compliance with FIPS PUB 186-3, “Digital Signature Standard”. Cryptographic signature functionality is performed by the cryptographic module operating in FIPS mode.	**RSA #1834**
	FCS_COP.1(3) Cryptographic Operation (cryptographic hashing)	SHA-1, SHA-256, and SHA512 cryptographic hashing implemented to meet FIPS PUB 180-3, “Secure Hash Standard”, are performed by the cryptographic module operating in FIPS mode.	**SHA #2934**
	FCS_COP.1(4) Cryptographic Operation (keyed-hash message authentication)	HMAC-SHA-1, HMAC-SHA-256, and HMAC-SHA512 keyed-hash message authentication, implemented to meet FIPS PUB 198-1, "The Keyed-Hash Message Authentication Code”, and FIPS PUB 180-3, “Secure Hash Standard”, are performed by the cryptographic module operating in FIPS mode.	**HMAC #2272**
	FCS_RBG_EXT.1 Cryptographic Operation (random bit generation)	CTR_DRBG (AES-256) random bit generation, implemented to meet NIST SP 800-90, is performed by the cryptographic module operating in FIPS mode.	**DRBG #910**

The TOE uses a software-based random number generator that complies with NIST Special Publication 800-90B when operating in the FIPS mode. It is seeded from the entropy source that provides a minimum of 256 bits of randomness accumulated from multiple software noise sources in the kernel. The noise sources are based on timing of various unpredictable system events. The amount of available entropy is constantly monitored and there are checks in place to guarantee that the random bit generator is seeded only when sufficient entropy is available.

The TOE generally fulfills all of the NIST SP 800-56B ‘Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography’ requirements, with the following table documenting specific conformance to the publication:

Table 24: NIST SP 800-56B implementation

NIST SP500-56B Section Reference	“should”, “should not”, or “shall not”	Implemented accordingly?	Rationale for deviation
5.6	Should	Yes	
5.8	Shall Not	Yes	
5.9	Shall Not (1st instance)	Yes	
5.9	Shall Not (2nd instance)	Yes	
6.1	Should Not	Yes	
6.1	Should (1st instance)	Yes	
6.1	Should (2nd instance)	Yes	
6.1	Should (3rd instance)	Yes	
6.1	Should (4th instance)	Yes	
6.1	Shall Not (1st instance)	Yes	
6.1	Shall Not (2nd instance)	Yes	
6.2.3	Should	Yes	
6.5.1	Should	Yes	
6.5.2	Should	Yes	
6.5.2.1	Should	Yes	
6.6	Shall Not	No	Not applicable
7.1.2	Should	Yes	
7.2.1.3	Should	Yes	
7.2.1.3	Should Not	No	Not applicable
7.2.2.3	Shall Not	Yes	
7.2.2.3	Should (1st instance)	Yes	
7.2.2.3	Should (2nd instance)	Yes	
7.2.2.3	Should (3rd instance)	Yes	
7.2.2.3	Should (4th instance)	Yes	
7.2.2.3	Should Not	Yes	
7.2.3.3	Should (1st instance)	Yes	
7.2.3.3	Should (2nd instance)	Yes	
7.2.3.3	Should (3rd instance)	Yes	
7.2.3.3	Should (4th instance)	Yes	
7.2.3.3	Should (5th instance)	Yes	
7.2.3.3	Should Not	No	Not applicable
8	Should	Yes	
8.3.2	Should Not	No	Not applicable

The TOE stores all persistent secret and private keys on a hard disk, and all derived and session keys in RAM. The TOE is designed to zeroize derived and session keys when they are no longer in use. The TOE implements commands to on-demand zeroize CSPs that can be invoked by an authorized administrator. The following table lists the applicable CSPs and identifies how they are zeroized:

Table 25: TOE CSPs

Identifier	Name	Generation / Algorithm	Purpose	Storage Location	Zeroization Summary
CSP1	TLS host keys	ANSI X9.31 / RSA	RSA key	Hard drive and RAM (plain text)	Keys are removed from RAM and overwritten once with zeroes when the forwarder process stops. An administrator can issues a command to wipe the keys from disk and overwrite once with zeroes the disk space where they are stored. Overwritten with 0x00
CSP2	TLS Session keys	ANSI X9.31 / AES-CBC	TLS Session keys - server to client, client to server	RAM (plain text)	Session keys are programmatically zeroized and new keys generated upon rekeying. Overwritten with 0x00.
CSP3	Diffie-Hellman key pair	ANSI x9.31 / DH	Key agreement for TLS sessions	RAM (plain text)	Cleared when device is powered down or as part of session termination. Overwritten by loss of capacitor charge in the memory cell.
CSP4	Username / Passwords	Secret	Critical security parameters used to authenticate the administrator login.	Hard drive (cipher text)	Passwords exist locally in the database. The passwords are stored in encrypted form only and cleared when no longer in use. Overwritten with 0x00.
				RAM (plain text)	Passwords in RAM are zeroized after creating or resetting the password. Overwritten with 0x00.
CSP5	PRNG Seed key	Entropy	Seed key for PRNG	RAM (plain text)	Cleared when device is powered down or overwritten by the new seed.

Identifier	Name	Generation / Algorithm	Purpose	Storage Location	Zeroization Summary
					Overwritten by loss of capacitor charge in the memory cell.

The TOE supports password based authentication and allows users to upload and associate with user identities certificates for public key client authentication.

The TOE implements HTTPS using TLS v1.0 protocol that complies with RFC 2246 with no extension. TLS is used to implement trusted channel and trusted path functionality.

The TLS 1.0 is implemented to support the following ciphersuites: TLS_RSA_WITH_AES_128_CBC_SHA, TLS_RSA_WITH_AES_256_CBC_SHA.

The TOE is capable of supporting other ciphers as part of standards-based protocol implementation but they are not enabled by default, and there is no administrative option to enable them.

7.3 User Data Protection

FDP_RIP.2

The TOE is designed to ensure its own integrity as well as to protect user data from unintentional reuse.

The TOE implements a residual information clearing mechanism as part of network packet processing. At the hardware level, packets are stored in the managed circular buffer where each slot allocates dedicated memory space of the exact size of the packet. Once the packet has been either transmitted or discarded, the memory used for that packet is returned back to the pool for reuse. Residual data is never transmitted from the TOE.

All plaintext secret and private cryptographic keys and CSPs used during TOE operation undergo a process whereby all data is zeroized when no longer used.

Cryptographic keys that are stored in the clear text are protected with restricted file permissions. There is no external interface available for viewing these keys. As such, the root administrator with a physical access is theoretically able to access these keys, however the assumption that TOE administrators are trusted, and physical access is secured.

7.4 Identification and Authentication

FIA_UIA_EXT.1, FIA_UAU_EXT.2, FIA_UAU.7, FIA_PMG_EXT.1

The TOE requires any user to be identified and authenticated before any other action, except to view the warning banner. The TOE does not allow unauthenticated configuration of the TOE’s services or features. The TOE allows unauthenticated network routing protocol traffic destined to the TOE, but does not include any management configuration of the TOE services.

The TOE authenticates a requesting user against their user name, password, and role. During the authentication process, password character entries are not echoed during a local session and are not transmitted in the clear during a remote session. The passwords can be composed of any combination of upper and lower case letters, numbers, and special characters and support passwords of 15 characters of greater.

For remote access via the Web Interface, only a secure HTTPS/TLS connection is allowed. Upon successful logon, users will be taken to the home screen of their pre-configured choice. Unsuccessful logon will result in a message stating the logon was unsuccessful. An administrator-specified number of failed logins will result in an account being locked.

Only authorized administrators are permitted access to the TOE using a local connection via the System Console. An Administrator must log in with a correct username and password. A correct login will enable access to the TOE via the System Console, and an unsuccessful logon will result in in a message stating the logon was unsuccessful.

The remote administrative interface has a configurable timeout, which by default is 15 minutes. After that period of inactivity, the session will be terminated. The local System Console has no such timeout period as it is assumed that there is sufficient physical security that access cannot be gained by unauthorized personnel.

The TOE uses a RBAC structure for restricting system access. Each authorized user is associated with an assigned role and specific permissions that determine their access to TOE features.

User role profiles are defined to provide two axes of permissions:

- Functions a role can access within the system
- Assets that are accessible for each type of function

The default roles for the TOE are as follows:

- **“Administrator” Role**: this role is used to manage the secure configuration and operation of the TOE
- **“Operator” Role**: this role can view and manipulate data, but cannot access or alter security functionality of the TOE.

Administrators can create users, associate passwords with user accounts, and can assign roles to a user. Administrators can access the TOE either via the Web Interface or through a System Console, while Operators are restricted to the Web Interface.

Local access to the System Console is for a set of administrative tasks, primarily intended for initial configuration or disaster recovery activities and its use is considered ‘maintenance mode’. The System Console allows access to a subset of administrative tasks available through the Web Interface. Administrators, after authentication, can execute various commands that implement the following administrative tasks:

- Change System Settings: Configure the network (DNS, firewall, proxy for packages, Domain, VPN, time, date and location, and hostname)
- Perform System Update

- Peer and Sensor Configuration: Sensor's listening interfaces, monitored networks, data source interface, and netflows generator
- Disaster Recovery Activities: Repair the database, reset/change admin password, reboot or restart services.
- Diagnostic Tools and Debugging: Check logs and network usage for debugging problems.

7.5 Security Management

FMT_MTD.1, FMT_SMF.1, FMT_SMR.2

The TOE supports RBAC, whereby a primary or root administrator – created automatically when installing an instance of the TOE – configures the access privileges of all other users of the system through the web interface.

The TOE can span multiple networks, and as such uses an asset structure for access control whereby asset objects are defined ranging from entire companies to a single IP address or group of hosts. This allows administrators to configure the system using abstractions as well allowing a deep specificity of role management. User role profiles are defined to provide permissions based on firstly the functions a user can access on the system and then the assets that are accessible for each type of function.

Roles in the TOE can be grouped into Administrator roles, whereby the security functionality of the product can be controlled to varying extents, and Users roles who have access to various levels of data depending on privileges, and can access data for monitoring and reporting but cannot otherwise alter security settings.

The TOE implements remote administrative access over HTTPS using the Web Interface or local access using System Console. To support remote administration, a management workstation with a web browser capable of HTTPS protocol-supporting TLS must be used. Local access requires physical access to the TOE via hardwired keyboard and mouse.

All sessions with all users are terminated upon restart of the TOE, or loss of network connectivity. There are no administrative actions possible prior to identification and authentication, as all users (including administrators) must log in prior to any action being possible.

7.6 Protection of the security functionality

FPT_APW_EXT.1, FPT_SKP_EXT.1, FPT_STM.1, FPT_TST.1, FPT_TUD.1

The TOE is a standalone appliance designed to function independently. As a result, both security functionality and measures to protect security functionality are focused on self-protection.

The TOE protects against tampering and unauthorized data disclosure by using dedicated communication channels protected by cryptographic means and access control methods. The TOE enforces user access controls and restricts access to the system and data to authorized and identified users.

The TOE protects stored passwords and cryptographic keys so they are not directly accessible in plaintext. There is no external interface available for viewing these keys. Locally stored password information is obscured by use of hashing (SHA256). Additionally, when login-related configuration information is typed through regular TOE interfaces it is obfuscated by displaying series of asterisks.

The TOE is a hardware appliance that implements a hardware-based real-time clock managed by an embedded OS, which also controls the exposure of administrative functions. This clock is used to produce reliable timestamps that are available for log accountability, synchronization with the operational environment, administrative session timeouts, and various cryptographic and protocol functionality. The TOE can be configured to synchronize with an external NTP server.

The TOE is based on a Debian “Wheezy” 7.8 platform running a Linux 3.4 kernel and implements a standard set of system start up self-tests. TOE performs both hardware and software self-tests.

Generally, self-test failures generate audit records. Some low-level critical failure modes can prevent TOE start-up or operation, and as a result will not generate audit records. In such cases, the TOE will enter failure mode that displays error codes that can be viewed via the console. The TOE can be configured to reboot or to stop with errors displayed when non-critical errors are encountered.

The cryptographic module performs self-tests during startup; the messages are displayed on the console and syslog records are generated for both successful and failed tests. When operating in FIPS mode, self-tests comply with the FIPS 140-2 requirements for self-testing. The module performs known-answer algorithm testing, integrity testing, and conditional self-tests. Upon failing any of its FIPS mode power-on self-tests, the TOE will refuse to boot. For all self-tests, successful completion of the set of self-tests is indicated by reaching the log-on prompt.

To ensure the correct startup and continued operation of the TOE, a process called "monit" performs the following checks:

- Check if the TOE processes are up and running. If a process is not running, it tries to re-start the process. If it cannot start the process, it logs the failure.
- Connect to the listening ports to check availability.
- Check if system memory or CPU is past a utilization threshold. If high usage is detected, create a log entry.

The TOE automatically checks for updates and notifies administrators when one is available. The update must be manually downloaded from an approved source, verified with a published hash, and applied using System Console interface. All updates are protected from tampering by the use of digital signatures. Updates are protected by a published hash to verify the integrity of the update. The TOE can utilize internal cryptographic module to perform hash verification functionality. The AlienVault website is the only authorized source of updates.

An administrator can verify the update through the administration interface, where the current version of the software installed and update version are displayed. In the event updates cannot be verified, administrators must not install and contact AlienVault support.

7.7 TOE access

FTA_SSL.3, FTA_SSL.4, FTA_SSL_EXT.1, FTA_TAB.1

The TOE allows remote administrative access using the Web Interface and local maintenance-mode access using the System Console. The TOE will display a customizable banner when a user initiates an interactive session either locally or remotely. The TOE can be configured to enforce an administrator-defined inactivity timeout, after which the inactive session is automatically terminated. Both Web interface and local System Console implement session timeout functionality, but use separate timeout settings that must be individually configured. Once a session (local or remote) has been terminated, the TOE requires the user to re-authenticate. The administrator can force termination of a current session by issuing the logout command.

7.8 Trusted path/channels

FTP_ITC.1, FTP_TRP.1

The TOE protects communication with various optional operational environment components, such as a Syslog Server, using TLS connection. If the negotiation of a secure session fails, a connection will not be established. To implement a secure channel to external entities the TOE uses a TLS v1.0 protocol with a public-key based authentication. A RSA host key pair can be generated elsewhere and imported into the TOE. Client RSA public keys have to be imported into the TOE.

The TOE ensures integrity and disclosure protection of a remote administrative sessions by implementing HTTPS /TLS. When a client attempts to connect using TLS, the TOE and the client will negotiate from the available list of mutually acceptable algorithms or cipher suites. In each case, AES-CBC with 128-bit or 256-bit keys is implemented for encryption and decryption and RSA, using up to 2048-bit keys, is implemented for key exchange and authentication. These methods provide identification of the external entity and prevent disclosure or undetected modification of data across the communication channel.

8 Acronyms and Terminology

8.1.1 *Acronyms*

The following table defines CC and Product specific acronyms used within this Security Target.

Table 26: Acronyms

Acronym	Definition
CC	Common Criteria
CSP	Critical Security Parameter
FIPS	Federal Information Processing Standard
HTTP	Hypertext Transfer Protocol
IP	Internet Protocol
IT	Information Technology
NIST	National Institute of Standards and Technology
OE	Operational Environment
OS	Operating System
OSP	Organizational Security Policy
PP	Protection Profile
RFC	Networking Working Group Request for Comment
SAR	Security Assurance Requirement
SFR	Security Functional Requirement
SSH	Secure Shell
ST	Security Target
TOE	Target of Evaluation
TSF	TOE Security Function
TSFI	TOE Security Function Interface

8.1.2 *Product Acronyms and Terminology*

The following table defines the CC and Product-specific terminology used within this Security Target.

Table 27: Terminology

Terminology	Definition
AAA	Authentication, Authorization, and Accounting (AAA). A security architecture for distributing systems for controlling remote access to services.
RADIUS	Remote Authentication Dial-In User Service (RADIUS) protocol that includes authentication and authorization.
RSA	Ron **R**ivest, Adi **S**hamir, Leonard **A**dleman. Public-key cryptosystem algorithm.
TACACS+	Terminal Access Controller Access-Control System Plus, an access control network protocol.