Avaya Virtual Services Platforms

Common Criteria Security Target
Document Version: 2.0

Prepared by:
Acumen Security
18504 Office Park Dr.
Montgomery Village, MD 20886

www.acumensecurity.net

Table of Contents

Revision History

Version	Date	Description
Version 2.0	July 2017	Updated for Assurance Continuity

1 Security Target Introduction

1.1 Security Target and TOE Reference

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

Category	Identifier
ST Title	Avaya Virtual Services Platform Security Target
ST Version	2.0
ST Date	July 2017
ST Author	Acumen Security, LLC.
TOE Identifier	Avaya Virtual Services Platform
TOE Hardware Version(s)	VSP 4000: VSP 4850GTS, VSP 4850GTS-PWR+, VSP 4450GSX-PWR+ VSP 8000: VSP 8284XSQ, VSP 8404
TOE Software Version	5.1.2.0
TOE OS	Mentor Graphics Linux 4.0
TOE Developer	Avaya, Inc.
Key Words	**Network device, Security Appliance**

Table 1 TOE/ST Identification

1.2 TOE Overview

The TOE consists of a family of Ethernet switches that can be deployed in different environments to suit the needs of varying networks. They can be deployed individually or in combination with other solutions. The TOE also provides network protection through the use of industry standard security functions.

1.3 TOE Description (Avaya VSP 4000 and VSP 8000)

The TOE consist of three (3) series of network switch families, the Avaya VSP 4000 and VSP 8000. The following sections provide an overview of the functionality provided by each appliance family and the physical characteristics of each platform within each family.

1.3.1 Virtual Services Platform 4000 Series: VSP 4850GTS, VSP 4850GTS-PWR+, VSP 4450GSX-PWR+

The VSP 4000 series are edge devices that are designed for small sites and delivers full-featured networking capabilities, while simplifying management by delivering multiple services without managing multiple protocols.

The following table identifies the physical characteristics of each appliance.

	VSP 4850GTS	VSP 4850GTS-PWR+	VSP 4450GSX-PWR+
Network Ports	• 48 x 10/100/1000BASE-T, • 2 x 1000BASE-SFP, • 2 x 10GBASE-SFP+.	• 48 x 10/100/1000BASE-T, • 2 x 1000BASE-SFP, • 2 x 10GBASE-SFP+.	• 36 x 100/1000BASE-SFP, • 12 x 10/100/1000BASE-T, • 2x 1000BASE-SFP , • 2 x 10GBASE-SFP+, • 2 x 10 Gigabit ports.
Enclosure	1 RU High		
Power Supply	• AC Input 300 W	• AC Input. Redundant 1000w	• AC input 1000W POE+
Processors	Freescale P2020	Freescale P2020	Freescale P2020

	VSP 4850GTS	VSP 4850GTS-PWR+	VSP 4450GSX-PWR+
Crypto Module	Mocana Cryptographic Suite B Module version 6.4.1f	Mocana Cryptographic Suite B Module version 6.4.1f	Mocana Cryptographic Suite B Module version 6.4.1f
Software	Version 5.1.2.0	Version 5.1.2.0	Version 5.1.2.0
TOE OS	Mentor Graphics Linux 4.0	Mentor Graphics Linux 4.0	Mentor Graphics Linux 4.0
Environment Requirements	• 0 to 50 degrees C • Humidity - 0 to 95% non-condensing		

Table 2: VSP 4000 series

1.3.2 Virtual Services Platform 8000 Series: VSP 8284XSQ (fixed configuration), VSP 8404 4-slot Switch

The VSP 8000 series platform offer flexibility by including versatile network connectivity and the latest-generation hardware. The compact form-factor is an innovation that better power efficiency and allows for an easier way to increase port density.

	VSP 8284XSQ	VSP 8404
Network Ports	• 80 x 10 Gigabit SFP+ • 4 x 40 Gigabit QSFP+	• 8408QQ: 8 x 40 Gigabit QSFP+, • 8418XSQ: 16 x 10 Gigabit SFP+ and 2 x 40 QSFP+, • 8424XS: 24 x 10 Gigabit SFP+, • 8424XT: 24 x 10 Gigabit RJ45.
Enclosure	2 RU High	
Power Supply	• AC Input 800 W • DC Input 800 W	
Processors	Freescale P2020	Freescale P2020
Crypto Module	Mocana Cryptographic Suite B Module version 6.4.1f	Mocana Cryptographic Suite B Module version 6.4.1f
Software	Version 5.1.2.0	Version 5.1.2.0
TOE OS	Mentor Graphics Linux 4.0	Mentor Graphics Linux 4.0
Environmental Requirements	• 0 to 50 degrees C • Humidity - 0 to 95% non-condensing	

Table 3: VSP 8000 series

1.4 TOE Evaluated Configuration

The TOE evaluated configuration consists of at least one of the following devices: VSP 4850GTS, VSP 4850GTS-PWR+, VSP 4450GSX-PWR+, VSP 8284XSQ, and/or VSP 8404. The evaluated configuration also supports the following external IT entities;

Component	Required	Usage/Purpose Description for TOE performance
Management Workstation through remote CLI and GUI	Yes	This includes any IT Environment Management workstation with an SSH client and web browser installed that is used by the TOE administrator to support TOE administration through HTTPS and SSH protected channels.
NTP Server	No	The TOE optionally supports communications with an NTP server to synchronize date and time.
Syslog Server	Yes	The syslog audit server is used for remote storage of audit records that have been generated by and transmitted from the TOE.
Certificate Authority	Yes	The CA is used in support of certificate validation operations.
OCSP Server	Yes	The OCSP server is used in support of certificate revocation testing.
AAA	Yes	This includes any IT environment AAA server that provides authentication services to TOE administration.

Table 4 IT Environment Components

1.5 TOE Architecture

1.5.1 Physical Boundaries

The TOE is a hardware and software solution that is comprised of the network device models described in section 1.3. The diagram below depicts the evaluated configuration. The red rectangle represents the physical boundary of the TOE.

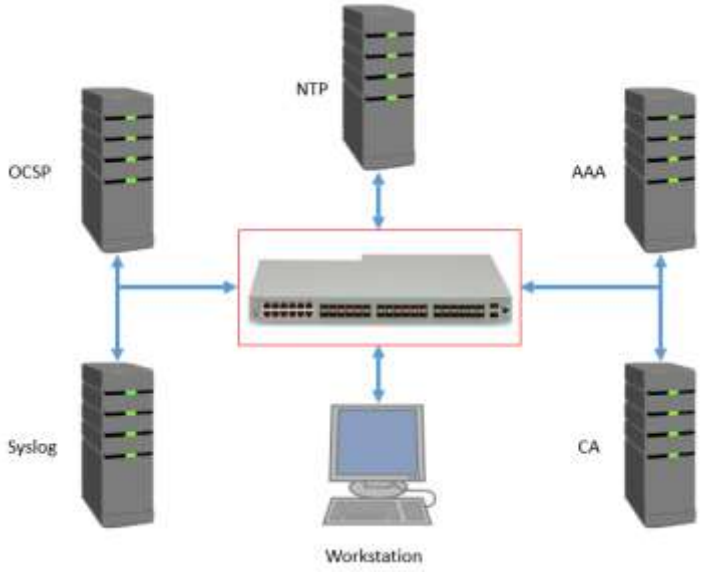

Figure 1 Physical Boundary

The IPv4 network on which the TOE resides is considered part of the environment. The software is pre-installed and is comprised of only the software versions identified in section 1. In addition, the software images are also downloadable from the Avaya website (https://support.avaya.com/) and are verified using digital signatures. An Avaya issued login and password are needed to access the secure download site. The guidance document posted on the NIAP website with this Security Target, are required for the evaluated configuration.

The TOE supports IPv6, however, the evaluated configuration was performed on an IPv4 network.

1.5.2 Logical Boundaries

The TOE provides several types of security functionalities, including.

- Security Audit
- Cryptography Support
- Identification & Authentication
- Security Management
- Protection of the TSF
- TOE Access
- Trusted Path/Channel

These features are described in more detail in the subsections below. In addition, the TOE implements all RFCs of the Collaborative Protection Profile for Network Devices necessary to satisfy testing/ assurance measures prescribed therein.

1.5.2.1 Security Audit

The Network Appliances provide extensive auditing capabilities. The TOE generates a comprehensive set of audit logs that identify specific TOE operations. For each event, the TOE records the date and time of each event, the type of event, the subject identity, and the outcome of the event. Auditable events include: failure on invoking cryptographic functionality such as establishment, termination and failure of a TLS session; establishment, termination and failure of an SSH session; establishment, termination and failure of an IPsec session; all use of the user identification mechanisms; any use of the authentication mechanism; any change in the configuration of the TOE, changes to time, initiation of TOE update, indication of completion of TSF self-test, termination of a remote session; and initiation and termination of a trusted channel.

The TOE is configured to transmit its audit messages to an external syslog server. Communication with the syslog server is protected using SSH.

The logs for all of the appliances can be viewed via the remote GUI interface or through the CLI. The records include the date/time the event occurred, the event/type of event, the user ID associated with the event, and additional information of the event and its success and/or failure.

1.5.2.2 Cryptographic Support

The TOE provides cryptographic support for the following features,

- TLS/HTTPS connectivity with the following entities:
    - Management Web Browser,
    - Audit Server.
- SSH connectivity with the following entities:

    - Management SSH Client.
- IPsec connectivity with the following entities:
    - AAA Server.
- Secure software update

The cryptographic services provided by the TOE are described below.

Cryptographic Method	Use within the TOE	CAVP Certificate #
RSA Signature Services	• Used in TLS session establishment • Used in SSH session establishment • Used in IPsec session establishment • Used in secure software update	2219
SP 800-90A CTR_DRBG	• Used in TLS session establishment • Used in SSH session establishment • Used in IPsec session establishment	1232
SHS	• Used to provide TLS traffic integrity verification • Used to provide SSH traffic integrity verification • Used to provide IPsec traffic integrity verification • Used in secure software update	3375
HMAC-SHS	• Used to provide TLS traffic integrity verification • Used to provide SSH traffic integrity verification • Used to provide IPsec traffic integrity verification	2679
AES	• Used to encrypt TLS traffic • Used to encrypt SSH traffic • Used to encrypt IPsec traffic	4100
SP 800-56A	• Used in TLS session establishment • Used in SSH session establishment • Used in IPsec session establishment	971
DSA	• Used in support of SP 800-56A	1140

Table 5: Provided Cryptography

1.5.2.3 Identification and Authentication

The TOE provides authentication services for administrative users to connect to the TOEs administrator interfaces (local CLI, remote CLI, and remote GUI). The TOE requires Authorized Administrators to authenticate prior to being granted access to any of the management functionality. In the Common Criteria evaluated configuration, the TOE is configured to require a minimum password length of 15 characters. The TOE provides administrator authentication against a local user database. Password-based authentication can be performed on any TOE administrative interface either locally or via an AAA server.

1.5.2.4 Security Management

The TOE provides secure administrative services for management of general TOE configuration and the security functionality provided by the TOE. Management can take place over a variety of interfaces including:

- Local console command line administration at each of the appliances;
- Remote command line administration via SSHv2 at each appliance;
- Remote GUI administration via HTTPS/TLS.

The TOE provides multiple interfaces to perform administration. While in the CLI command mode, the user has access to six distinct modes that provide a specific set of commands. Higher modes can mostly access commands of the lower modes, except, if they conflict with commands of the current mode. The CLI modes are as follows;

- User EXEC Mode: Initial mode of access.
- Privileged EXEC Mode: User mode and password combination determines access level.
- Global Configuration Mode: Use this mode to make changes to the running configuration.
- Interface Configuration Mode: Use this mode to modify or configure logical interface, VLAN or a physical interface.
- Router Configuration Mode: Use this mode to modify a protocol.
- Application Configuration Mode: Use this mode to access the applications.

The TOE also offers a web-based graphical user interface in order to securely manage the appliances. This is known as the Enterprise Device Manager (EDM) and is accessible once it has been enable through the CLI.

All administration functions can be accessed via, remote CLI, remote GUI or via a direct connection to the TOE. The TOE provides the ability to securely manage the below listed functions;

- All TOE administrative users;
- All identification and authentication;
- All audit functionality of the TOE;
- All TOE cryptographic functionality;
- The timestamps maintained by the TOE;
- Update to the TOE.

The TOE supports the configuration of login banners to be displayed at time of login and inactivity timeouts to terminate administrative sessions after a set period of inactivity.

1.5.2.5 Protection of the TSF

The TOE protects against interference and tampering by untrusted subjects by implementing identification, authentication, and access controls to limit configuration to Administrators. The TOE prevents reading of cryptographic keys and passwords.

The TOE internally maintains the date and time. This date and time is used as the timestamp that is applied to audit records generated by the TOE. Administrators can update the TOE's clock manually, or can configure the TOE to use NTP to synchronize the TOE's clock with an external time source. Finally, the TOE performs testing to verify correct operation of the security appliances themselves. The TOE verifies all software updates via digital signature (2048-bit RSA/SHA-256) and requires administrative intervention prior to the software updates being installed on the TOE to avoid the installation of unauthorized software.

1.5.2.6 TOE Access

The TOE can terminate inactive sessions after an Authorized Administrator configurable time period. Once a session has been terminated the TOE requires the user to re-authenticate to establish a new session. The TOE can also display an Authorized Administrator specified banner on both the CLI and GUI management interfaces prior to allowing any administrative access to the TOE.

1.5.2.7 Trusted Path/Channels

The TOE supports several types of secure communications, including,

- Trusted paths with remote administrators over SSH,
- Trusted paths with remote administrators over TLS/HTTPS,
- Trusted channels with remote IT environment audit servers over TLS,
- Trusted channels with remote IT environment AAA servers over IPsec.

2 Conformance Claims

2.1 CC Conformance

This TOE is conformant to:

- Common Criteria for Information Technology Security Evaluations Part 1, Version 3.1, Revision 4, September 2012
- Common Criteria for Information Technology Security Evaluations Part 2, Version 3.1, Revision 4, September 2012: Part 2 extended
- Common Criteria for Information Technology Security Evaluations Part 2, Version 3.1, Revision 4, September 2012: Part 3 conformant

2.2 Protection Profile Conformance

This TOE is conformant to:

- Collaborative Protection Profile for Network Devices, Version 1.0, 27 February 2015 [NDcPP].

2.3 Conformance Rationale

This Security Target provides exact conformance to Version 1.0 of the Collaborative Network Device Protection Profile. The security problem definition, security objectives and security requirements in this Security Target are all taken from the Protection Profile performing only operations defined there.

3 Security Problem Definition

The security problem definition has been taken from [NDcPP] and is reproduced here for the convenience of the reader.

3.1 Threats

The threats for the Network Device are grouped according to functional areas of the device in the sections below.

3.1.1 Communications with the Network Device

A network device communicates with other network devices and other network entities. The endpoints of this communication can be geographically and logically distant and may pass through a variety of other systems. The intermediate systems may be untrusted providing an opportunity for unauthorized communication with the network device or for authorized communication to be compromised. The security functionality of the network device must be able to protect any critical network traffic (administration traffic, authentication traffic, audit traffic, etc.). The communication with the network device falls into two categories: authorized communication and unauthorized communication.

Authorized communication includes network traffic allowable by policy destined to and originating from the network device as it was designed and intended. This includes critical network traffic, such as network device administration and communication with an authentication or audit logging server, which requires a secure channel to protect the communication. The security functionality of the network device includes the capability to ensure that only authorized communications are allowed and the capability to provide a secure channel for critical network traffic. Any other communication is considered unauthorized communication.

The primary threats to network device communications addressed in this cPP focus on an external, unauthorized entity attempting to access, modify, or otherwise disclose the critical network traffic. A poor choice of cryptographic algorithms or the use of non-standardized tunneling protocols along with weak administrator credentials, such as an easily guessable password or use of a default password, will allow a threat agent unauthorized access to the device. Weak or no cryptography provides little to no protection of the traffic allowing a threat agent to read, manipulate and/or control the critical data with little effort. Nonstandardized tunneling protocols not only limit the interoperability of the device but lack the assurance and confidence standardization provides through peer review.

3.1.1.1 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.

3.1.1.2 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.

3.1.1.3 T.UNTRUSTED_COMMUNICATION_CHANNELS

Threat agents may attempt to target network devices that do not use standardized secure tunneling 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.

3.1.1.4 T.WEAK_AUTHENTICATION_ENDPOINTS

Threat agents may attempt to target network devices that do not use standardized secure tunneling 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.

3.1.2 Valid Updates

Updating network device software and firmware is necessary to ensure that the security functionality of the network device is maintained. The source and content of an update to be applied must be validated by cryptographic means; otherwise, an invalid source can write their own firmware or software updates that circumvents the security functionality of the network device. Methods of validating the source and content of a software or firmware update by cryptographic means typically involve cryptographic signature schemes where hashes of the updates are digitally signed.

Unpatched versions of software or firmware leave the network device susceptible to threat agents attempting to circumvent the security functionality using known vulnerabilities. Non-validated updates or updates validated using non-secure or weak cryptography leave the updated software or firmware vulnerable to threat agents attempting to modify the software or firmware to their advantage.

3.1.2.1 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.

3.1.3 Audited Activity

Auditing of network device activities is a valuable tool for administrators to monitor the status of the device. It provides the means for administrator accountability, security functionality activity reporting, reconstruction of events, and problem analysis. Processing performed in response to device activities may give indications of a failure or compromise of the security functionality. When indications of activity that impact the security functionality are not generated and monitored, it is possible for such activities to occur without administrator awareness. Further, if records are not generated and retained, reconstruction of the network and the ability to understand the extent of any compromise could be negatively affected. Additional concerns are the protection of the audit data that is recorded from alteration or unauthorized deletion. This could occur within the TOE, or while the audit data is in transit to an external storage device.

Note this cPP requires that the network device generate the audit data and have the capability to send the audit data to a trusted network entity (e.g., a syslog server).

3.1.3.1 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.

3.1.4 Administrator and Device Credentials Data

A network device contains data and credentials which must be securely stored and must appropriately restrict access to authorized entities. Examples include the device firmware, software, configuration authentication credentials for secure channels, and administrator credentials. Device and administrator keys, key material, and authentication credentials need to be protected from unauthorized disclosure and modification. Furthermore, the security functionality of the device needs to require default authentication credentials, such as administrator passwords, be changed.

Lack of secure storage and improper handling of credentials and data, such as unencrypted credentials inside configuration files or access to secure channel session keys, can allow an attacker to not only gain access to the network device, but also compromise the security of the network through seemingly authorized modifications to configuration or though man-in-the-middle attacks. These attacks allow an unauthorized entity to gain access and perform administrative functions using the Security Administrator's credentials and to intercept all traffic as an authorized endpoint. This results in difficulty in detection of security compromise and in reconstruction of the network, potentially allowing continued unauthorized access to administrator and device data.

3.1.4.1 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 include replacing existing credentials with an attacker's credentials, modifying existing credentials, or obtaining the administrator or device credentials for use by the attacker.

3.1.4.2 T.PASSWORD_CRACKING

Threat agents may be able to take advantage of weak administrative passwords to gain privileged access to the device. Having privileged access to the device provides the attacker unfettered access to the network traffic, and may allow them to take advantage of any trust relationships with other network devices.

3.1.5 Device Failure

Security mechanisms of the network device generally build up from roots of trust to more complex sets of mechanisms. Failures could result in a compromise to the security functionality of the device. A network device self-testing its security critical components at both start-up and during run-time ensures the reliability of the device's security functionality.

3.1.5.1 T.SECURITY_FUNCTIONALITY_FAILURE

A component of the network device may fail during start-up or during operations causing a compromise or failure in the security functionality of the network device, leaving the device susceptible to attackers.

3.2 Assumptions

This section describes the assumptions made in identification of the threats and security requirements for network devices. The network device is not expected to provide assurance in any of these areas, and as a result, requirements are not included to mitigate the threats associated.

3.2.1 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 and/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 will not include any requirements on physical tamper protection or other physical attack mitigations. The cPP will 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. [OE.PHYSICAL]

3.2.2 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 computing platform for general purpose applications (unrelated to networking functionality). [OE.NO_GENERAL_PURPOSE]

3.2.3 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 for particular types of network devices (e.g, firewall). [OE.NO_THRU_TRAFFIC_PROTECTION]

3.2.4 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 being 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. [OE.TRUSTED_ADMIN]

3.2.5 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. [OE.UPDATES]

3.2.6 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. [OE.ADMIN_CREDENTIALS_SECURE]

3.3 Organizational Security Policy

An organizational security policy is a set of rules, practices, and procedures imposed by an organization to address its security needs. For the purposes of this cPP a single policy is described in the section below.

3.3.1 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. [FTA_TAB.1]

4 Security Objectives

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

4.1 Security Objectives for the Operational Environment

The following subsections describe objectives for the Operational Environment.

4.1.1 OE.PHYSICAL

Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.

4.1.2 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.

4.1.3 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.

4.1.4 OE.TRUSTED_ADMIN

TOE Administrators are trusted to follow and apply all guidance documentation in a trusted manner.

4.1.5 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.

4.1.6 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.

5 Security Requirements

This section identifies the Security Functional Requirements for the TOE and/or Platform. The Security Functional Requirements included in this section are derived from Part 2 of the Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 4, dated: September 2012 and all international interpretations.

5.1 Conventions

The CC defines operations on Security Functional Requirements: assignments, selections, assignments within selections and refinements. This document uses the following font conventions to identify the operations defined by the CC:

- Assignment: Indicated with *italicized* text;
- Refinement made by PP author: Indicated with **bold** text and ~~strikethroughs~~, if necessary;
- Selection: Indicated with underlined text;
- Assignment within a Selection: Indicated with *italicized and underlined text*;
- Iteration: Indicated by appending the iteration number in parenthesis, e.g., (1), (2), (3).
- Where operations were completed in the PP itself, the formatting used in the PP has been retained.

Explicitly stated SFRs are identified by having a label 'EXT' after the requirement name for TOE SFRs. Formatting conventions outside of operations matches the formatting specified within the PP.

5.2 TOE Security Functional Requirements

This section identifies the Security Functional Requirements for the TOE. The TOE Security Functional Requirements that appear below in Table 1 are described in more detail in the following subsections.

Requirement	Auditable Events	Additional Audit Record Contents
FAU_GEN.1	None.	None.
FAU_GEN.2	None.	None.
FAU_STG_EXT.1	None.	None.
FCS_CKM.1	None.	None.
FCS_CKM.2	None.	None.
FCS_CKM.4	None	None
FCS_COP.1(1)	None.	None.
FCS_COP.1(2)	None.	None.
FCS_COP.1(3)	None.	None.
FCS_COP.1(4)	None.	None.
FCS_RBG_EXT.1	None.	None.
FIA_PMG_EXT.1	None.	None.
FIA_UIA_EXT.1	All use of the identification and authentication mechanism.	Provided user identity, origin of the attempt. (e.g., IP address)
FIA_UAU_EXT.2	All use of the identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UAU.7	None.	None
FIA_X509_EXT.1	Unsuccessful attempt to validate a certificate	Reason for failure
FIA_X509_EXT.2	None.	None.
FIA_X509_EXT.3	None.	None.

Requirement	Auditable Events	Additional Audit Record Contents
FMT_MOF.1(1)/Trusted Update	Any attempt to initiate a manual update	None.
FMT_MTD.1	All management activities of TSF data.	None.
FMT_SMF.1	None.	None.
FMT_SMR.2	None.	None.
FPT SKP EXT.1	None.	None.
FPT APW EXT.1	None.	None.
FPT_STM.1	Changes to the time.	The old and new values for the time. Origin of the attempt (e.g., IP address).
FPT TUD EXT.1	Initiation of update; result of the update attempt (success or failure)	No additional information.
FPT TST EXT.1	None.	
FTA_SSL_EXT.1	Any attempts at unlocking of an interactive session.	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.
FTA_TAB.1	None.	None.
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.
FCS_SSHS_EXT.1	Failure to establish an SSH Session	Reason for failure
	Successful SSH rekey	Non-TOE endpoint of a connection (IP Address)
FCS_TLSS_EXT.1	Failure to establish a TLS session	Reason for failure
FCS_TLSC_EXT.1	Failure to establish a TLS session	Reason for failure
FCS_HTTPS_EXT.1	Failure to establish an HTTPS Session.	Reason for failure
FCS_IPSEC_EXT.1	Failure to establish an IPsec SA.	Reason for failure

Table 6: TOE SFR and Auditable Events

5.2.1 Class: Security Audit (FAU)

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 comprising:*
    a. *Administrative login and logout (name of user account shall be logged if individual user accounts are required for administrators).*
    b. *Security related configuration changes (in addition to the information that a change occurred it shall be logged what has been changed).*

c. *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).*
d. *Resetting passwords (name of related user account shall be logged).*
e. *Starting and stopping services (if applicable)*
f. [no other actions];

d) [Specifically defined auditable events listed in Table 6].

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 Table* 6].

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.

FAU_STG_EXT.1 External Audit Trail 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.

FAU_STG_EXT.1.3 The TSF shall [drop new audit data] when the local storage space for audit data is full.

5.2.2 Class: Cryptographic Support (FCS)

FCS_CKM.1 Cryptographic Key Generation (For Asymmetric Keys)

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

- RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3;
- FFC schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.1

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

FCS_CKM.2 Cryptographic Key Establishment (Refined)

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

- RSA-based key establishment schemes that meets the following: NIST Special Publication 800-56B, "Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography";
- Finite field-based key establishment schemes that meets the following: NIST Special Publication 800-56A, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography"

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

FCS_CKM.4 Cryptographic Key Destruction

FCS_CKM_EXT.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method *[selection:*

- *For volatile memory, the destruction shall be executed by a single direct overwrite* [consisting of zeroes] *followed by a read-verify.*
    - *If the read-verification of the overwritten data fails, the process shall be repeated again.*
- *For non-volatile flash memory, the destruction shall be executed by* [a single, direct overwrite consisting of zeroes] *followed by a read-verify.*
    - *If the read-verification of the overwritten data fails, the process shall be repeated again.*

] that meets the following: *No Standard.*

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

FCS_COP.1.1(1) The TSF shall perform *encryption/decryption* in accordance with a specified cryptographic algorithm *AES used in [*CBC*] 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].

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

FCS_COP.1.1(2) The TSF shall perform *cryptographic signature services (generation and verification)* in accordance with a specified cryptographic algorithm [

- RSA Digital Signature Algorithm and cryptographic key sizes **(modulus)** *[2048 bits]*,

]

that meets the following: [

- For RSA schemes: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSAPKCS2v1_5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3].

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 [SHA-1, SHA-256] ~~and cryptographic key sizes [assignment: *cryptographic key sizes*]~~ that meet the following: *ISO/IEC 10118-3:2004*.

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-SHA-1, HMAC-SHA-256] and cryptographic key sizes [*160 bits, 256 bits*] **and message digest sizes [160 bits, 256 bits]** that meet the following: *ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2"*.

FCS_RBG_EXT.1 Extended: Cryptographic Operation (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 The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [*[4]* software-based noise source] with a minimum of [128 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", of the keys and hashes that it will generate.

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.

FCS_HTTPS_EXT.1.3 The TSF shall establish the connection only if [the peer initiates handshake].

Note: NIAP TD 125 is used to update FCS_HTTPS_EXT.1.3 to reflect that the implementation of an HTTPS server that does not require mutual authentication.

FCS_IPSEC_EXT.1 Extended: IPSEC

FCS_IPSEC_EXT.1.1 The TSF shall implement the IPsec architecture as specified in RFC 4301.

FCS_IPSEC_EXT.1.2 The TST shall have a nominal, final entry in the SPD that matches anything that is otherwise unmatched, and discards it.

FCS_IPSEC_EXT.1.3 The TSF shall implement transport mode and [no other mode].

FCS_IPSEC_EXT.1.4 The TSF shall implement the IPsec protocol ESP as defined by RFC 4303 using the cryptographic algorithms AES-CBC-128, AES-CBC-256 (both specified by RFC 3602) and [no other algorithms] together with a Secure Hash Algorithm (SHA)-based HMAC.

FCS_IPSEC_EXT.1.5 The TSF shall implement the protocol: [

- IKEv1, using Main Mode for Phase 1 exchanges, as defined in RFCs 2407, 2408, 2409, RFC 4109, [no other RFCs for extended sequence numbers], and [no other RFCs for hash functions];
- IKEv2 as defined in RFC 5996 and [with no support for NAT traversal], and [no other RFCs for hash functions]]

FCS_IPSEC_EXT.1.6 The TSF shall ensure the encrypted payload in the [IKEv1, IKEv2] protocol uses the cryptographic algorithms AES-CBC-128, AES-CBC-256 as specified in RFC 3602 and [no other algorithm].

FCS_IPSEC_EXT.1.7 The TSF shall ensure that [

- IKEv1 Phase 1 SA lifetimes can be configured by an Security Administrator based on [
    - length of time, where the time values can configured within [*1193046*] hours;]*;*
- IKEv2 SA lifetimes can be configured by an Security Administrator based on [
    - length of time, where the time values can configured within [*1193046*] hours;]

FCS_IPSEC_EXT.1.8 The TSF shall ensure that [

- IKEv1 Phase 2 SA lifetimes can be configured by a Security Administrator based on [
    - length of time, where the time values can be configured within [*1193046*] hours;];
- IKEv2 Child SA lifetimes can be configured by a Security Administrator based on [

- length of time, where the time values can be configured within [*1193046*] hours;]

FCS_IPSEC_EXT.1.9 The TSF shall generate the secret value x used in the IKE Diffie-Hellman key exchange ("x" in g^x mod p) using the random bit generator specified in FCS_RBG_EXT.1, and having a length of at least [*4096*] bits.

FCS_IPSEC_EXT.1.10 The TSF shall generate nonces used in [IKEv1, IKEv2] exchanges of length [

- at least 128 bits in size and at least half the output size of the negotiated pseudorandom function (PRF) hash] .

FCS_IPSEC_EXT.1.11 The TSF shall ensure that all IKE protocols implement DH Groups 14 (2048-bit MODP), and [no other DH groups].

FCS_IPSEC_EXT.1.12 The TSF shall be able to ensure by default that the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the [IKEv1 Phase 1, IKEv2 IKE_SA] connection is greater than or equal to the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the [IKEv1 Phase 2, IKEv2 CHILD_SA] connection.

FCS_IPSEC_EXT.1.13 The TSF shall ensure that all IKE protocols perform peer authentication using [*RSA*] that use X.509v3 certificates that conform to RFC 4945 and [Pre-shared Keys].

FCS_IPSEC_EXT.1.14 The TSF shall only establish a trusted channel to peers with valid certificates.

FCS_SSHS_EXT.1 Extended: SSH Server

FCS_SSHS_EXT.1.1 The TSF shall implement the SSH protocol that complies with RFCs 4251, 4252, 4253, 4254, and [no other RFCs].

FCS_SSHS_EXT.1.2 The TSF shall ensure that the SSH protocol implementation supports the following authentication methods as described in RFC 4252: public key-based, password-based.

FCS_SSHS_EXT.1.3 The TSF shall ensure that, as described in RFC 4253, packets greater than [*256*] bytes in an SSH transport connection are dropped.

FCS_SSHS_EXT.1.4 The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms and rejects all other encryption algorithms: *aes128-cbc, aes256-cbc, [*no other algorithms*]*.

FCS_SSHS_EXT.1.5 The TSF shall ensure that the SSH transport implementation uses [ssh-rsa] *and* [no other public key algorithms] as its public key algorithm(s) and rejects all other public key algorithms.

FCS_SSHS_EXT.1.6 The TSF shall ensure that the SSH transport implementation uses [hmac-sha1] *and* [HMAC-SHA2-256] as its MAC algorithm(s) and rejects all other MAC algorithm(s).

FCS_SSHS_EXT.1.7 The TSF shall ensure that [diffie-hellman-group14-sha1] *and* [no other methods] are the only allowed key exchange methods used for the SSH protocol.

FCS_SSHS_EXT.1.8 The TSF shall ensure that the SSH connection be rekeyed after no more than 2^28 packets have been transmitted using that key.

FCS_TLSC_EXT.1 Extended: TLS Client

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

- *Mandatory Ciphersuites:*
    - *TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*
- [TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268
- TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246
- TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246].

FCS_TLSC_EXT.1.2 The TSF shall verify that the presented identifier matches the reference identifier according to RFC 6125.

FCS_TLSC_EXT.1.3 The TSF shall only establish a trusted channel if the peer certificate is valid.

FCS_TLSC_EXT.1.4 The TSF shall present the Supported Elliptic Curves Extension in the Client Hello with the following NIST curves: ***[secp256r1, secp384r1]*** and no other curves.

FCS_TLSS_EXT.1 Extended: TLS Server

FCS_TLSS_EXT.1.1 The TSF shall implement [TLS 1.2 (RFC 5246)] supporting the following ciphersuites:

- *Mandatory Ciphersuites:*
    - *TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*
- [TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268
- TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246
- TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246].

FCS_TLSS_EXT.1.2 The TSF shall deny connections from clients requesting SSL 1.0, SSL 2.0, SSL 3.0, TLS 1.0, and [TLS 1.1].

FCS_TLSS_EXT.1.3 The TSF shall generate key establishment parameters using RSA with key size 2048 bits and [no other size].

5.2.3 Class: Identification and Authentication (FIA)

FIA_PMG_EXT.1 Password Management

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

1. *Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters:* "!", "@", "#", "$", "%", "^", "&", "*", "(", ")", "'", "+", "-", ".", "/", ":", ";", "<", "=", ">", "?", "[", "\", "]", "^", "_", "`", "{", "|", "}", and "~". *;*
2. *Minimum password length shall settable by the Security Administrator, and shall support passwords of 15 characters or greater;*

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;
- [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

FIA_UAU_EXT.2 Extended: Password-based Authentication Mechanism

FIA_UAU_EXT.2.1 The TSF shall provide a local password-based authentication mechanism, [*remote password-based authentication mechanism*] to perform administrative user authentication.

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.

FIA_X509_EXT.1 X.509 Certificate Validation

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

- RFC 5280 certificate validation and certificate path validation.
- The certificate path must terminate with a trusted CA certificate.
- The TSF shall validate a certificate path by ensuring the presence of the basicConstraints extension and that the CA flag is set to TRUE for all CA certificates.
- The TSF shall validate the revocation status of the certificate using [the Online Certificate Status Protocol (OCSP) as specified in RFC 2560].
- 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 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 TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.
    - *[OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.]*

FIA_X509_EXT.1.2 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.

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 [IPsec, SSH], 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 [accept the certificate].

FIA_X509_EXT.3 X.509 Certificate Requests

FIA_X509_EXT.3.1 The TSF shall generate a Certificate Request Message as specified by RFC 2986 and be able to provide the following information in the request: public key and [Common Name, 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 Class: Security Management (FMT)

FMT_MOF.1(1)/TrustedUpdate Management of security functions behavior

FMT_MOF.1.1(1)/TrustedUpdate The TSF shall restrict the ability to enable the functions *to perform manual update* to *Security Administrators*.

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*

FMT_SMF.1 Specification of Management Functions

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

- *Local and remote administration of the TOE and the services provided by the TOE via the TOE CLI/GUI, as described above;*
- *Ability to configure the access banner*
- *Ability to configure the session inactivity time before session termination or locking*
- *Ability to update the TOE, and to verify the updates using digital signature capability prior to installing those updates*
- *The ability to manage the cryptographic functionality which allows the Authorized Administrator the ability to identify and configure the algorithms used to provide protection of the data, such as generating RSA keys.;*
- [No other capabilities]

FMT_SMR.2 Restrictions on Security Roles

FMT_SMR.2.1 The TSF shall maintain the roles:

- *User EXEC Mode: Initial mode of access.*
- *Privileged EXEC Mode: User mode and password combination determines access level.*
- *Global Configuration Mode: Use this mode to make changes to the running configuration.*
- *Interface Configuration Mode: Use this mode to modify or configure logical interface, VLAN or a physical interface.*
- *Router Configuration Mode: Use this mode to modify a protocol.*
- *Application Configuration Mode: Use this mode to access the applications.*

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

FMT_SMR.2.3 The TSF shall ensure that the conditions

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

are satisfied.

5.2.5 Class: Protection of the TSF (FPT)

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.

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.

FPT_STM.1 Reliable Time Stamps

FPT_STM.1.1 The TSF shall be able to provide reliable time stamps.

FPT_TUD_EXT.1 Extended: 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 as well as the most recently installed version of the TOE firmware/software.

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 [published hash] prior to installing those updates.

FPT_TST_EXT.1: TSF Testing

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

- Software integrity test,
- AES Known Answer Test,
- SHS Known Answer Test,
- HMAC Known Answer Test,
- DRBG Known Answer Test,
- RSA Known Answer Test].

5.2.6 Class: TOE Access (FTA)

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.

FTA_SSL.3 TSF-initiated Termination

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

FTA_SSL.4 User-initiated Termination

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

FTA_TAB.1 Default TOE Access Banners

FTA_TAB.1.1 **Refinement:** 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**.

5.2.7 Class: Trusted Path/Channels (FTP)

FTP_ITC.1 Inter-TSF trusted channel

FTP_ITC.1.1 The TSF shall be **capable of using** [IPsec, TLS] **to** provide a trusted communication channel between itself and **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 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 [audit logging, remote authentication].

FTP_TRP.1 Trusted Path

FTP_TRP.1.1 The TSF shall **be capable of using** [SSH, TLS, HTTPS] **to** provide a communication path between itself and **authorized 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 provides detection of modification of the channel data*.

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

FTP_TRP.1.3 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 Collaborative Protection Profile for Network Devices contains all the requirements claimed in this Security Target. 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 Collaborative Protection Profile for Network Devices which are derived from Common Criteria Version 3.1, Revision 4. The assurance requirements are summarized in the table below.

Assurance Class	Components	Components 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 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

Table 7: Security Assurance Requirements

6 TOE Summary Specification

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

#	TOE SFR	Rationale
1	FAU_GEN.1	The switches provide extensive auditing capabilities. The TOE generates a comprehensive set of audit logs that identify specific TOE operations. For each event, the TOE records the date and time of each event, the type of event, the subject identity, and the outcome of the event. The types of events that cause audit records to be generated include identification and authentication related events, and administrative events. Each of the events is specified in the audit record is in enough detail to identify the user for which the event is associated (e.g. user identity, MAC address, IP address), when the event occurred, where the event occurred, the outcome of the event, and the type of event that occurred. The logs for all of the appliances can be viewed via the remote GUI interface or through the CLI (local or remote). Additionally, the TOE supports remote audit logging using the syslog standard with an external server. Audit messages are entered into the log and the subset of the log contents are sent to the syslog server. When an administrative command is executed, the TOE sets up the session data structure which includes the “user identity”. When an audit log is generated, the session data is passed along with the audit information and the TOE extracts the “user identity” from the session data structure. The TOE generates the following types of audit logs during operation: • Start-up of the TOE from both cold boot and reboot, • Shutdown of the TOE (when shut down from the CLI), ○ Note: The audit functionality of the TOE cannot be separately shutdown and started. Startup and shutdown of auditing is only facilitated by starting and stopping the switch • All administrative actions (both security relevant and non-security relevant) from the local CLI, Remote CLI, and GUI, • IKE/IPsec session establishment with the syslog server, • IKE/IPsec session closure with the syslog server, • Errors during IKE/IPsec session establishment (e.g., algorithm mismatch), • Remote administrative HTTPS/TLS connection establishment , • Remote administrative HTTPS/TLS connection closure, • Errors during Remote administrative HTTPS/TLS connection establishment, • Remote administrative SSH connection establishment, • Remote administrative SSH connection closure, • Errors during Remote administrative SSH connection establishment, • Generation of self-signed certificates, • Import of certificates, • Deletion of certificates, • Successful authentication attempts (from the local CLI, Remote CLI, and GUI), • Unsuccessful authentication attempts (from the local CLI, Remote CLI, and GUI), • Unsuccessful certificate validation for the presence of the basicConstraints extension missing, • Unsuccessful certificate validation for the CA flag is set to TRUE for all CA certificates, • Unsuccessful certificate validation for trust chain verification failure, • Unsuccessful certificate validation for revocation status, • All attempts to update the TOE software, • Changes to time, • Start of a local administrative session, • End of a local administrative session, • Administration session timeout (from the local CLI, Remote CLI, and GUI). Below you will see several examples of audit record on the TOE,

#	TOE SFR	Rationale
		CP1 [07/22/16 03:14:19.746:UTC] 0x000d8602 00000000 GlobalRouter SSH INFO SSH user authentication succeeded for user acumensec on host 192.168.128.35 CP1 [07/22/16 03:14:19.763:UTC] 0x00004620 00000000 GlobalRouter SNMP INFO SSH new session login CP1 [07/22/16 03:14:19.767:UTC] 0x000d8602 00000000 GlobalRouter SSH INFO SSH CLI session start: user acumensec on host 192.168.128.35 CP1 [07/22/16 03:14:19.773:UTC] 0x00004620 00000000 GlobalRouter SNMP INFO SSH new session login CP1 [07/22/16 03:14:20.772:UTC] 0x00030656 00000000 GlobalRouter SW INFO user acumensec logged in through ssh,Unsuccessful Login attempts from last login is:0 and Last Successful Login time is:Fri Jul 22 03:07:49 2016 CP1 [07/22/16 03:14:46.206:UTC] 0x000d8602 00000000 GlobalRouter SSH INFO SSH CLI session end: user acumensec on host 192.168.128.35 CP1 [07/22/16 03:14:46.206:UTC] 0x000d8602 00000000 GlobalRouter SSH INFO SSH session closed by user acumensec on host 192.168.128.35
2	FAU_GEN.2	The TOE ensures that each auditable event is associated with the user that triggered the event. For example, a human user, user identity or related session ID would be included in the audit record. For an IT entity or device, the IPv4 address, MAC address, host name, or other configured identification is included in the audit record. The audit record is generated with the required information and stored plaintext on the device.
3	FAU_STG_EXT.1	The TOE is configured to export syslog records to a specified, external syslog server. The TOE protects communications with an external syslog server via syslog messages encrypted via TLS over TCP (RFC 5425). When communicating with an external syslog server, the TOE acts as a TLS client. The TOE then periodically initiates a connection with the syslog server. Once the server has accepted the TLS connection as a TLS server, the TOE pushes the audit logs to the syslog server over the secure channel. The maximum size of audit records stored by the TOE can be configured by an administrator. The upper limit on local audit storage is based on the amount of available hard drive space, but an administrator can set a lower limit if desired. For audit records stored internally to the TOE the audit records are stored in a log file. The TOE stops recording new audit records when the audit trail becomes full. Only Authorized Administrators are able to clear the local logs, and local audit records are stored in a directory that does not allow administrators to modify the contents. When the TOE is configured for syslog server, syslog records are transferred immediately. If the connection with the server is broken or the syslog server is full the TOE will not retransmit the syslog records after normal connectivity is resumed.
4	FCS_CKM.1	The TOE can create a RSA public-private key pair with a RSA key size of 2048 bits. The RSA algorithm implementation is provided the included Mocana cryptographic library. The RSA key pair can be used to generate a Certificate Signing Request (CSR). In support of FFC Diffie-Hellman the TOE supports DSA. The relevant CAVP certificate numbers are listed in Table 6.
5	FCS_CKM.2	In support of secure cryptographic protocols, the TOE supports key establishment schemes, including, • FFC Diffie-Hellman as specified in NIST SP 800-56A, • RSA Key Transport as specified in SP NIST 800-56B. The TOE is fully compliant to both SP 800-56A and SP 800-56B. The switches implement each “shall” statement in each standard and do not implement any “shall not” statements in either of the standards.
6	FCS_CKM.4	The TOE actively performs a destruction of keys and Critical Security Parameters (CSPs) when no longer required for use. The switches store several types of keys in volatile memory in plaintext, including, • SSH Session Encryption Key, • SSH Session Integrity Key, • TLS Session Encryption Key, • TLS Session Integrity Key, • IKE Session Encryption Key, • IKE Session Integrity Key, • IPsec Session Encryption Key, • IPsec Session Integrity Key,

#	TOE SFR	Rationale
		Each plaintext key stored in volatile memory is associated with a cryptographic session. In each instance, after the session closes, the key is overwritten with the value “00” After the overwrite operation is complete, the TOE performs a specific "read-verify" operation to confirm that the storage space no longer contains the key. The switches do not store any keys in non-volatile storage in plaintext form.
7	FCS_COP.1(1)	The TOE provides symmetric encryption and decryption capabilities using AES in CBC mode (128, 256 bits) as described AES as specified in ISO 18033-3. AES is implemented in support of the following protocols: IPSEC, TLS, and SSH. The relevant CAVP certificate numbers are listed in Table 6.
8	FCS_COP.1(2)	The TOE provides cryptographic signature services using RSA Digital Signature Algorithm with key size of 2048 as specified in FIPS PUB 186-4, “Digital Signature Standard”. The relevant CAVP certificate numbers are listed in Table 6.
9	FCS_COP.1(3)	The TOE provides cryptographic hashing services using SHA-1 and SHA-256 as specified in FIPS Pub 180-4 “Secure Hash Standard.” SHS hashing is used within several services including, IKE/IPsec hashing, TLS/HTTPS, and SSH. SHA-256 is used in conjunction with RSA signatures for verification of software image integrity. The relevant CAVP certificate numbers are listed in Table 6.
10	FCS_COP.1(4)	The TOE provides keyed-hashing message authentication services using HMAC-SHA-1 and HMAC-SHA-256. The product supports the following cryptographic parameters for MACing, as specified in ISO/IEC 9797-2:2011: • Key length: 256-bits, 512-bits, • Hash function used: Sha-1, SHA-256, • Block size: 256-bits, 512-bits, • Output MAC: 160-bits, 256-bits. The relevant CAVP certificate numbers are listed in Table 6.
11	FCS_RBG_EXT.1	The TOE implements a NIST-approved AES-CTR Deterministic Random Bit Generator (DRBG), as specified in SP 800-90. The relevant CAVP certificate numbers are listed in Table 6. The TOE implements a random bit generator in support of various cryptographic operations, including, RSA key establishment schemes, Diff-Hellman key establishment schemes, TLS session establishment, SSH session establishment, and IPsec session establishment. The entropy source used to seed the Deterministic Random Bit Generator (e.g. based on SP 800-90A/B/C) is a random set of bits or bytes that are regularly supplied to the DRBG by polling four different set of software sources in threads. All entropy is continuously health tested by the DRBG as per the tests defined in section 11.3 of SP 900-90A before being used as a seed. Any initialization or system errors during bring-up or processing of this system causes a reboot resulting in the DRBG being reseeded.
12	FCS_IPSEC_EXT.1	The TOE uses IPsec in support of communications with external AAA servers. The TOE exclusively supports transport mode, allowing for only the payload of the packet to be encrypted. Since the TOE does not support tunnel mode, any external connections that attempt to negotiate a tunnel mode IPsec session will fail. The SPD is implemented by using Protect and Discard of the packets. A SPD set can contain multiple entries, each with a different access list. The SPD entries are searched in a sequence - the TOE attempts to match the packet to the access list (acl) specified in that entry. When a packet matches a permit entry in a particular access list, the method of security in the corresponding crypto map is applied. The traffic matching the permit acls would then flow through the IPsec tunnel and be classified as “PROTECTED”. Traffic that matches a discard acl on the interface would be DISCARDED. Traffic that does not match a permit or discard acl would be allowed to BYPASS the tunnel. For example, not having a permit or reject acl for icmp would allow ping traffic to flow unencrypted if a permit crypto map was not configured that matches the ping traffic. A final reject all traffic acl can be applied to an interface to reject all other traffic not explicitly allowed. The TOE implements IPsec to provide certificate-based authentication and encryption services to prevent unauthorized viewing or modification of data as it travels over the external network to the remote AAA server. The TOE implementation of the IPsec standard (in accordance with the RFCs noted in the SFR) uses the Encapsulating Security Payload (ESP) protocol to provide authentication, encryption and anti-replay services. The IKE protocol (IKEv1 and IKEv2) implements Peer Authentication using the RSA algorithm with X.509v3 certificates. During this authentication, the distinguished name (DN) is verified mto ensure the certificate is valid and is from a valid entity. The DN naming attributes in the certificate is compared with the expected DN naming attributes and deemed valid if the attribute types are the same and the values are the same and as expected.

#	TOE SFR	Rationale
		IKEv1 separates negotiation into two phases: phase 1 and phase 2. Phase 1 creates the first encrypted channel, which protects later the phase 2 negotiation messages. The key negotiated in phase 1 enables IKE peers to communicate securely in phase 2. During Phase 2 IKE establishes the IPsec SA. IKEv2 also supports two separates negotiations: IKE SAs and Child SAs. IKE SAs create the first encrypted channel, which protects later the Child SA messages. This is similar to IKEv1 negotiation. IKE (both IKEv1 and IKEv2) maintains a trusted channel, referred to as a Security Association (SA), between the TOE and the remote AAA server that is also used to manage IPsec connections, including: • The negotiation of mutually acceptable IPsec options between peers (including peer authentication parameters, • The establishment of additional Security Associations to protect packets flows using Encapsulating Security Payload (ESP), and • The agreement of secure bulk data encryption AES keys for use with ESP. After the two peers agree upon a policy, the security parameters of the policy are identified by an SA established at each peer, and these IKE SAs apply to all subsequent IKE traffic during the negotiation. The TOE supports IKEv1 session establishment. As part of this support, the TOE explicitly does not support aggressive mode for IKEv1 exchanges and only uses main mode negotiation. For IKEv1, the TOE supports configuration lifetimes of both Phase 1 SAs and Phase 2 SAs based on the life time of the ST. The default time value for Phase 1 SAs is 24 hours and the maximum time is 1193046 hours. The default time value for Phase 2 SAs is also 24 hours and may also be configured to a maximum of 1193046 hours. Similar to IKEv1, for IEKv2 the TOE supports configuration lifetimes of both IKE SAs and Child SAs based on the life time of the ST. The default time value for Phase 1 SAs is 24 hours and the maximum time is 1193046 hours. The default time value for Phase 2 SAs is also 24 hours and may also be configured to a maximum of 1193046 hours. The TOE provides AES-CBC-128, and AES-CBC-256 for encrypting the IKE (both IKEv1 and IKEv2) payloads. The TOE supports Diffie-Hellman Group 14 (2048-bit keys), in support of both IKE versions. Key Establishment. These keys are generated using the AES-CTR Deterministic Random Bit Generator (DRBG), as specified in SP 800-90, and the following corresponding key sizes (in bits) are used: 320 (for DH Group 14) bits. The TOE generates the secret value 'x' used in the IKEv1 and IKEv2 Diffie-Hellman key exchange ('x' in gx mod p) using the NIST approved DRBG specified in FCS_RBG_EXT.1 and having possible lengths of 4096 bits. When a random number is needed for a nonce, the probability that a specific nonce value will be repeated during the life a specific IPsec SA is less than 1 in 2128. All nonces are generated using the AES-CTR DRBG. The TOE implementation of the IPsec standard (in accordance with the RFCs noted in the SFR and using cryptographic algorithms AES-CBC-128 and AESCBC-256 together with HMAC-SHA1, HMAC-SHA-256) uses the Encapsulating Security Payload (ESP) protocol to provide authentication, encryption and anti-replay services. During the SA negotiations, if a non-match is encountered, the process stops and an error message is received.
13	FCS_TLSC_EXT.1	In support of secure communication with external entities, the TOE supports the TLS protocol acting as a TLS client. TLS is used to facilitate communication with the following entities, • Syslog Servers In support of these connections, the TOE only support TLS 1.2. No other TLS protocol versions, such as, TLS 1.0, TLS 1.1 or SSL 3.0 are offered. The following ciphersuites are supported for communications with syslog servers: • TLS_RSA_WITH_AES_128_CBC_SHA • TLS_RSA_WITH_AES_256_CBC_SHA, • TLS_RSA_WITH_AES_128_CBC_SHA256, • TLS_RSA_WITH_AES_256_CBC_SHA256. The switches support full validation of the presented TLS server certificates, including, Common Name, DNS Name, URI Name, IP addresses, wildcards and Service Name. The specific reference identifier is configured as part of the external syslog configuration using the “syslog host” command. The “server-cert-name” parameter (the CN) is input and used as the reference ID. The connection is only valid when presented that certificate from the configured IP address. The TOE does not support certificate pinning for TLS clientconnection.

#	TOE SFR	Rationale
14	FCS_HTTPS_EXT.1	In support of secure communication with external entities, the TOE supports the TLS protocol acting as a TLS server. TLS is used to facilitate communication with the following entities, • Remote administrators The communication with remote administrators is over a TLS-protected HTTPS connection. In support of these connections, the TOE only support TLS 1.2. Connections using other version of TLS or SSL, such as, TLS 1.0, TLS 1.1 or SSL 3.0 are actively denied by the TOE. The following ciphersuites are supported for communications with remote administrators: • TLS_RSA_WITH_AES_128_CBC_SHA • TLS_RSA_WITH_AES_256_CBC_SHA, • TLS_RSA_WITH_AES_128_CBC_SHA256, • TLS_RSA_WITH_AES_256_CBC_SHA256. All other proposed ciphersuites are denied. Because the TOE does not support DHE_RSA, the switches do not ever send a server key exchange message.
15	FCS_TLSS_EXT.1	
16	FCS_SSHS_EXT.1	The TOE uses SSH for to facilitate secure remote administrative sessions (CLI). The TOE's SSH implementation supports the following, • Use of 2048-bit RSA keys in support of SSH_RSA for public key-based authentication; • Dropping SSH packets greater than 256 bytes. This is accomplished by buffering all data for a particular SSH packet transmission until the buffer limit is reached and then dropping the packet; • Strict compliance with RFCs 4251, 4252, 4253, and 4254, ○ No optional options included in the RFCs have been implemented; • Encryption algorithms AES-CBC-128 and AES-CBC-256 to ensure confidentiality of the session; • Password based authentication; • Public key based authentication; • Hashing algorithm hmac-sha1 and hmac-sha2-256 ensure the integrity of the session; • Enforcement of DH Group 14 as the only allowed key exchange method.
17	FIA_PMG_EXT.1	The TOE supports the local definition of users with corresponding passwords. The passwords can be composed of any combination of upper and lower case letters, numbers, and special characters (that include: “!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, “)”, “””, “+”, “-“, “.”, “/”, “:”, “;”, “<”, “=”, “>”, “?”, “[“, “\”, “]”, “^”, “_”, “`”, “{“, “|”, “}”, and “~”. The minimum password length is settable by the Authorized Administrator. When the TOE is configured for "Common Criteria Compliance" the minimum password length is set to 15 characters. In addition to locally managed passwords, the TOE support communications with remote AAA servers. These communications are facilitated over a secure IPsec connection.
18	FIA_UIA_EXT.1 FIA_UAU_EXT.2	The TOE requires all users to be successfully identified and authenticated before allowing any TSF mediated actions to be performed. Administrative access to the TOE is facilitated through one of several interfaces, • Directly connecting to the TOE • Remotely connecting via SSHv2 • Remotely connecting to the GUI via HTTPS/TLS Regardless of the interface at which the administrator interacts, the TOE will enforce username and password authentication or public-key based authentication. Only after the administrative user presents the correct authentication credentials will access to the TOE administrative functionality be granted. No access is allowed to the administrative functionality of the TOE until an administrator is successfully identified and authenticated. The TOE provides a local password based authentication mechanism and authentication via integration with a remote AAA server. The process for authentication is the same for administrative access whether administration is occurring via direct connection or remotely. At initial login, the administrative user is prompted to provide a username. After the user provides the username, the user is prompted to provide the administrative password associated with the user account. The TOE then either grant administrative access (if the combination of username and password is correct) or indicate that the login was unsuccessful. The TOE does not provide a reason for failure in the cases of a login failure.
19	FIA_UAU.7	For all authentication, regardless of the interface, the TOE displays only "*" characters when the administrative password is entered so that the password is obscured.

#	TOE SFR	Rationale
20	FIA_X509_EXT.1 FIA_X509_EXT.2 FIA_X509_EXT.3	The TOE uses X.509v3 certificates as defined by RFC 5280 to support the following connections, • IPsec connections with external AAA servers, • TLS connections with external syslog servers. The TOE creates Certificate Signing Request (CSRs). These signing requests contain the following fields, • Public Key • Common Name • Country This signing request is then sent to a CA for generation of a CA signed certificate. The TOE supports the following methods to obtain a certificate from a CA: • Simple Certificate Enrollment Protocol (SCEP) Each local certificate is digitally signed providing protection from unauthorized modification. If a certificate is modified in any way, it would be invalidated and rendered useless. The digital signature verifications process would show that the certificate had been tampered with when the hash value would be invalid. The certificate chain establishes a sequence of trusted certificates, from a peer certificate to the root CA certificate. Within the PKI hierarchy, all enrolled peers can validate the certificate of one another if the peers share a trusted root CA certificate or a common subordinate CA. Each CA corresponds to a trust point. The X.509 certificates are validated using the certificate path validation algorithm defined in RFC 5280, which can be summarized as follows: • The public key algorithm and parameters are checked • The current date/time is checked against the validity period revocation status is checked • Issuer name of X matches the subject name of X+1 • Name constraints are checked • Policy OIDs are checked • Policy constraints are checked; issuers are ensured to have CA signing bits • Path length is checked • Critical extensions are processed The obtains certificates from a SCEP server for IPSec and TLS server. In order to verify the revocation status of the presented certificates Online Certificate Status Protocol (OCSP) is used. The physical security of the TOE (A.PHYSICAL_PROTECTION) protects the switch and the certificates from being tampered with or deleted. In addition, the TOE identification and authentication security functions protect an unauthorized user from gaining access to the TOE. If the connection to determine the certificate validity cannot be established, the TOE accepts the certificate.
21	FMT_MOF.1(1)/ Trusted Update	The TOE does not provide automatic updates to the software version running on the TOE. The Security Administrators (a.k.a Authorized Administrators) can query the software version running on the TOE, and can initiate updates to (replacements of) software images. When software updates are made available, the Authorized Administrators can obtain, verify the integrity of, and install those updates. This verification uses digital signatures.
22	FMT_MTD.1	The TOE provides the ability for Security Administrators (a.k.a Authorized Administrators) to access TOE data, such as audit data, configuration data, security attributes, session thresholds and updates. Access to this data is governed by the privileges assigned to the administrative users. None of this functionality is accessible prior to the administrator logging into the TOE. The term “Authorized Administrator” is used in this ST to refer to any of the predefined user privilege levels.
23	FMT_SMF.1	The TOE provides all the capabilities necessary to securely manage the TOE. The Security Administrators (a.k.a Authorized Administrators) user can connect to the TOE using the CLI to perform these functions via remote CLI over SSHv2, at the local console, or via remote GUI over an HTTPS connection. The specific management capabilities available from the TOE include: • Local and remote administration of the TOE and the services provided by the TOE via the TOE CLI/GUI, as described above, • Ability to configure the access banner, • Ability to configure the session inactivity time before session termination or locking, • Ability to update the TOE, and to verify the updates using digital signature capability prior to installing

#	TOE SFR	Rationale
		those updates, The ability to manage the cryptographic functionality which allows the Authorized Administrator the ability to identify and configure the algorithms used to provide protection of the data, such as generating RSA keys. While the TOE supports configuration of IKE/IPsec via the TOE GUI the evaluated configuration only configures IKE/IPsec via the TOE CLI
24	FMT_SMR.2	The TOE supports multiple administrative roles when accessing the administrative interface through the local or remote CLI. These roles define the access that is allowed per role. The following list identifies the configuration capabilities assigned to each role. • User EXEC Mode: Initial mode of access. • Privileged EXEC Mode: User mode and password combination determines access level. • Global Configuration Mode: Use this mode to make changes to the running configuration. • Interface Configuration Mode: Use this mode to modify or configure logical interface, VLAN or a physical interface. • Router Configuration Mode: Use this mode to modify a protocol. • Application Configuration Mode: Use this mode to access the applications. Additionally, the TOE supports one authorized user role when accessing the TOE through the remote GUI interface. This role has full access to the TOE management capabilities defined in the NDcPP.
25	FPT_SKP_EXT.1	All keys and pre-shared keys are stored on the TOE are protected from unauthorized modification and substitution. The TOE stores symmetric keys only in volatile memory never on persistent media. The TOE admin interface does not provide any mechanism to view sensitive data (passwords or keys) once stored. Unauthenticated operators do not have write access to modify, change, or delete keys. The TOE encrypts and stores all private keys in a secure directory that is not readily accessible to administrators; therefore, there is no administrative interface access provided to directly manipulate the keys.
26	FPT_APW_EXT.1	No passwords are ever stored as clear text. Passwords are stored on the TOE in a secured partition in non-plaintext. Prior to writing on disks each password is hashed (SHA-256) with a salt. During subsequent authentication attempts passwords entered are converted into a SHA-256 digest using a salt value. This is compared to the digest value for that user stored in the secured partition. Access is only granted if the values match.
27	FPT_TST_EXT.1	The TOE runs a suite of self-tests during initial start-up to verify its correct operation. If any of the tests fail, the Authorized Administrator will have to log into the TOE to determine which test failed and why. During the system bootup process (power on or reboot), the TOE performs various Power on Startup Tests (POSTs) for the cryptographic components of the TOE. During initialization and self-test execution, the module inhibits all access to the cryptographic algorithms. Additionally, the power-on self-tests are performed after the cryptographic systems are initialized but prior to the underlying OS initialization of external interfaces; this prevents the security appliances from passing any data before completing selftests. In the event of a power-on self-test failure, the cryptographic module will force the platform to reload and reinitialize the operating system and cryptographic components. This operation ensures no cryptographic algorithms can be accessed unless all power on self-tests are successful. These tests include: • AES Known Answer Test - For the encrypt test, a known key is used to encrypt a known plain text value resulting in an encrypted value. This encrypted value is compared to a known encrypted value to ensure that the encrypt operation is working correctly. The decrypt test is just the opposite. In this test a known key is used to decrypt a known encrypted value. The resulting plaintext value is compared to a known plaintext value to ensure that the decrypt operation is working correctly. • HMAC Known Answer Test - For each of the hash values listed, the HMAC implementation is fed known plaintext data and a known key. These values are used to generate a MAC. This MAC is compared to a known MAC to verify that the HMAC and hash operations are operating correctly. • RNG/DRBG Known Answer Test - For this test, known seed values are provided to the DRBG implementation. The DRBG uses these values to generate random bits. These random bits are compared to known random bits to ensure that the DRBG is operating correctly. • SHA Known Answer Test – For each of the values listed, the SHA implementation is fed known data and key. These values are used to generate a hash. This hash is compared to a known value to verify they match and the hash operations are operating correctly. • RSA Signature Known Answer Test (both signature/verification) - This test takes a known plaintext value and Private/Public key pair and used the public key to encrypt the data. This value is compared to a known encrypted value to verify that encrypt operation is working properly. The encrypted data is then decrypted using the private key. This value is compared to the original plaintext value to ensure the decrypt operation is working properly.

#	TOE SFR	Rationale
		• Software Integrity Test - This test is run automatically whenever the system images is loaded and confirms through use of digital signature verification that the image file that's about to be loaded was properly signed and maintained its integrity since being signed. The system image is digitally signed prior to being made available for download from Avaya.
28	FPT_TUD_EXT.1	Authorized Administrator can query the software version running on the TOE, and can initiate updates to software images. When software updates are made available, an administrator can obtain, verify the integrity of via digital signature, and install those updates. The updates can be downloaded from https://support.avaya.com. The TOE image files are digitally signed so their integrity can be verified during the boot process, and an image that fails an integrity check will not be loaded. The public keys used by the update verification mechanism are contained on the TOE. The TOE compares the update files' signature using a certificate that comes pre-loaded on the device and is stored in the trust store. As part of the build process, the update image is signed with the Avaya private key. This is done using an RSA 2048/SHA-256 digital signature. Only if the signature/hash is correct, will the image be installed. If an update is unsuccessful, a message is delivered to the user. Since the update process attempts to update a different copy than what is currently being run, the current active image remains the same and the user continues to run the same code that was being run before the upgrade attempt was made.
29	FPT_STM.1	The TOE provides a source of date and time information used in audit event timestamps. The clock function is reliant on the system clock provided by the underlying hardware. The TOE can optionally be set to receive clock updates from an NTP server. 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.
30	FTA_SSL_EXT.1	The TOE provides the administrative user to defined inactivity time out periods for administrative sessions. The inactivity period for CLI (local and remote) and GUI (remote) administrative access are maintained separately and are configured separately through the TOE administrative interfaces.
	FTA_SSL.3	If an administrative session remains inactive for the configured length of time, the administrative session is terminated. After termination, administrative authentication is required to access any of the administrative functionality of the TOE. This is applicable from both local and remote administrative sessions.
31	FTA_SSL.4	An Authorized Administrator is able to exit out of both local and remote administrative sessions. When accessing the TOE via the CLI (both local and remote), the exit command is used. When accessing the TOE via the remote GUI, the logout button is used.
32	FTA_TAB.1	For TOE administration, the GUI (TLS/HTTPS), CLI (SSH) and local console CLI are available. Prior to an administrative user authenticating, that user is presented with an access display banner which displays an advisory notice and consent warning message regarding unauthorized use of the TOE. This banner will be displayed prior to allowing Administrator access through those interfaces.
33	FTP_ITC.1	The TOE protects communications with authorized IT entities via IPsec or TLS, as follows: • Trusted channels with audit servers are protected via TLS, • Trusted channels with AAA servers are protected via IPsec. This protects the data from disclosure by encryption and by checksums that verify that data has not been modified.
34	FTP_TRP.1	All remote administrative communications take place over a secure encrypted session. Remote CLI connections take place over an SSHv2 tunnel. The SSHv2 session is encrypted using AES encryption. Remote GUI connections take place over a TLS/HTTPS connection. The TLS session is encrypted using AES encryption. The remote administrators are able to initiate both SSHv2 and TLS/HTTPS communications with the TOE. The TOE rejects all insecure remote authentication attempts (e.g., telnet and HTTP).

Table 8: TOE SFR descriptions

7 Acronyms

Acronym	Description
AAA	Authentication, Authorization and Accounting
CA	Certificate Authority
NTP	Network Time Protocol
OCSP	Online Certificate Status Protocol
SSH	Secure Shell
TOE	Target of Evaluation
TLS	Transport Layer Security
VSP	Virtual Services Platform

End of Document