Hewlett Packard Enterprise Moonshot-180XGc, 45XGc, 45Gc Switch Modules (NDPP11e3) Security Target

Version 0.3
02/05/16

Prepared for:

Hewlett Packard Enterprise

153 Taylor Street
Littleton, MA 01460-1407

Prepared By:

Gossamer Laboratories

www.gossamersec.com

LIST OF TABLES

1. Security Target Introduction

This section identifies the Security Target (ST) and Target of Evaluation (TOE) identification, ST conventions, ST conformance claims, and the ST organization. The TOE is Hewlett Packard Enterprise Moonshot-180XGc, 45XGc, 45Gc Switch Modules provided by Hewlett Packard Enterprise. The TOE is being evaluated as a network infrastructure device.

The Security Target contains the following additional sections:

- Conformance Claims (Section 2)
- Security Objectives (Section 3)
- Extended Components Definition (Section 4)
- Security Requirements (Section 5)
- TOE Summary Specification (Section 6)

Conventions

The following conventions have been applied in this document:

- Security Functional Requirements – Part 2 of the CC defines the approved set of operations that may be applied to functional requirements: iteration, assignment, selection, and refinement.
    - Iteration: allows a component to be used more than once with varying operations. In the ST, iteration is indicated by a letter placed at the end of the component. For example FDP_ACC.1a and FDP_ACC.1b indicate that the ST includes two iterations of the FDP_ACC.1 requirement, a and b.
    - Assignment: allows the specification of an identified parameter. Assignments are indicated using bold and are surrounded by brackets (e.g., [**assignment**]). Note that an assignment within a selection would be identified in italics and with embedded bold brackets (e.g., [***[selected-assignment]***]).
    - Selection: allows the specification of one or more elements from a list. Selections are indicated using bold italics and are surrounded by brackets (e.g., [***selection***]).
    - Refinement: allows the addition of details. Refinements are indicated using bold, for additions, and strike-through, for deletions (e.g., "… **all** objects …" or "… ~~some~~ **big** things …").
- The NDPP uses an additional convention – the 'case' – which defines parts of an SFR that apply only when corresponding selections are made or some other identified conditions exist. Only the applicable cases are identified in this ST and they are identified using **bold** text.
- Other sections of the ST – Other sections of the ST use bolding to highlight text of special interest, such as captions.

1.1 Security Target Reference

ST Title – Hewlett Packard Enterprise Moonshot-180XGc, 45XGc, 45Gc Switch Modules (NDPP11e3) Security Target

ST Version – Version 0.3

ST Date – 02/05/16

1.2 TOE Reference

TOE Identification Hewlett Packard Enterprise Moonshot-180XGc, 45XGc, 45Gc Switch Modules

TOE Developer – Hewlett Packard Enterprise

Evaluation Sponsor – Hewlett Packard Enterprise

1.3 TOE Overview

The Target of Evaluation (TOE) is Hewlett Packard Enterprise Moonshot-180XGc, 45XGc, 45Gc Switch Modules. The Moonshot Switches are switch appliances that provide network connectivity for the following: Cloud computing, service providers, Web2.0, health care, Universities, Government agencies and for use in HPE enclosures. The Moonshot Switches include the HPE Comware V7.1 network operating system, which delivers enterprise grade resiliency and is designed for data center convergence with full support for IEEE Data Center Bridging (DCB) for lossless Ethernet, and Fibre Channel over Ethernet (FCoE) protocols. The switches support IETF industry standard TRILL (Transparent Interconnection of Lots of Links) that enables loop free large Layer 2 networks with multi-path support. The switch provides Intelligent Resilient Framework (IRF) which enables multiple switches to be virtualized and managed as a single entity with HPE's Intelligent Management Center (IMC). The IMC is not within the scope of the evaluation. Management of the IRF group can and should occur via any of the IRF group members by an authorized administrator using the CLI.

In the evaluated configuration, the switches can be deployed as a single switch device or alternately as a group of up to four devices connected using the HPE Intelligent Resilient Framework (IRF) technology to effectively form a logical switch device. The IRF technology requires that devices be directly connected to one another using an IRF stack using one or more dedicated Ethernet connections that are used to coordinate the overall logical switch configuration and also to forward applicable network traffic as necessary between attached devices. The IRF technology does not require that switches be co-located, but can be attached using standard LACP for automatic load balancing and high availability. Note that the IRF connections are not secured (e.g., using encryption) by the TOE, so the IRF group members must be collocated and the IRF connections need to be as protected as the IRF group devices themselves.

The Moonshot Switches support uplink modules and plug-in modules, which provide additional functionality (e.g., various numbers and types of network connection ports). All of the available plug-in modules are included in the evaluated configuration (see below).

1.4 TOE Description

The HPE Moonshot-180XGc, 45XGc, 45Gc Switch Modules are a Gigabit Ethernet switch appliances that consists of hardware and software components. The software used is Comware V7.1 and is common code base of a modular nature with only the modules applicable for the specific hardware installed.

The following modules, extending the physically available ports, are supported by the HPE Moonshot-180XGc, 45XGc, 45Gc Switch Modules and can optionally be used since they do not affect any of the claimed security functions but rather serve to extend available network connectivity:

- HPE Moonshot–4QSFP+ Uplink Module
- HPE Moonshot-16SFP+ Uplink Module
- HPE Moonshot-6SFP+ Uplink Module

1.4.1 TOE Architecture

The HPE Moonshot-180XGc, 45XGc, 45Gc Switch Modules comprising the TOE includes a common software code base, called Comware. Comware is special purpose appliance system software that implements a wide array of networking technology, including: IPv4 dual-stacks, a data link layer, layer 2 and 3 routing, Ethernet switching, VLANs, Intelligent Resilient Framework (IRF) routing, Quality of Service (QoS), etc. The evaluated version of Comware is V7.1. It should be noted that Comware runs on a variety of underlying architectures including

VxWorks, Linux,pSOS and Windows; however, the only underlying architecture found in the evaluated configuration is Linux.

Comware V7.1 implements full modularization and multi-process applications, as well as provides the following benefits:

- Full modularization—Brings improvements in system availability, virtualization, multi-core multi-CPU applications, distributed computing, and dynamic loading and upgrading.
- Openness—Comware V7.1 is a generic, open system based on Linux.
- Improved operations—Comware V7.1 improves some detailed operations. For example, it uses preemptive scheduling to improve real-time performance.

Comware V7.1 optimizes the following functions:

- Virtualization—Supports N:1 virtualization.
- ISSU—Supports ISSU for line cards.
- Auxiliary CPU and OAA—Improve scalability for devices.

Comware V7.1 comprises four planes: management plane, control plane, data plane, and infrastructure plane. Each is summarized below:

- Infrastructure plane – The infrastructure plane provides basic Linux services and Comware support functions. Basic Linux services comprise basic Linux functions, C language library functions, data structure operations, and standard algorithms. Comware support functions provide software and service infrastructures for Comware processes, including all basic functions.
- Data plane – The data plane provides data forwarding for local packets and received IPv4 packets at different layers
- Control plane – The control plane comprises all routing, signaling, and control protocols, such as MPLS, OSPF, and security control protocols. It generates forwarding tables for the data plane.
- Management plane – The management plane provides a management interface for administrators and operators to configure, monitor, and manage Comware V7.1. The management interface comprises a CLI accessed using SSH.

From a security perspective, the TOE implements NIST-validated cryptographic algorithms that support the IPsec and SSH protocols as well as digital signature services that support the secure update capabilities of the TOE. Otherwise, the TOE implements a wide range of network switching protocols and functions.

1.4.1.1 Physical Boundaries

A TOE device (HPE Moonshot-180XGc, 45XGc, 45Gc Switch Modules) is a modular switch appliance with a fixed number of ports and modular uplink module.

The TOE can be configured to rely on and use a number of other components in its operational environment.

- Syslog server – to receive audit records when the TOE is configured to deliver them to an external log server.
- RADIUS and TACACS+ servers – The TOE can be configured to use external authentication servers.
- Management Workstation – The TOE supports CLI access and as such an administrator would need an SSHv2 client to use the administrative interface

1.4.1.2 Logical Boundaries

This section summarizes the security functions provided by HPE Moonshot-180XGc, 45XGc, 45Gc Switch Modules:

- Security audit
- Cryptographic support
- User data protection
- Identification and authentication
- Security management
- Protection of the TSF
- TOE access
- Trusted path/channels

1.4.1.2.1 Security audit

The TOE is designed to be able to generate logs for a wide range of security relevant events. The TOE can be configured to store the logs locally so they can be accessed by an administrator or alternately to send the logs to a designated syslog server.

1.4.1.2.2 Cryptographic support

The TOE includes NIST-validated cryptographic mechanisms that provide key management, random bit generation, encryption/decryption, digital signature and secure hashing and key-hashing features in support of higher level cryptographic protocols, including IPsec and SSHv2. Note that in the evaluated configuration, the TOE must be configured in FIPS mode to ensure that CAVP tested cryptographic functions are used.

1.4.1.2.3 User data protection

The TOE supports a wide variety of network access control functions. While implementing its network access control functions, the TOE is carefully designed to ensure that it doesn't inadvertently reuse network or management data. This is accomplished primarily by clearing and zero-padding of memory structures and packet buffers when allocated.

1.4.1.2.4 Identification and authentication

The TOE requires users (i.e., administrators) to be successfully identified and authenticated before they can access any security management functions available in the TOE. The TOE offers both a locally connected console and a network accessible interface (SSHv2) for interactive administrator sessions.

The TOE supports the local (i.e., on device) definition of administrators with usernames and passwords. Additionally, the TOE can be configured to use the services of trusted RADIUS and TACACS+ servers in the operational environment to support, for example, centralized user administration.

1.4.1.2.5 Security management

The TOE provides Command Line (CLI) commands (locally via a serial console or remotely via SSH) to access the available functions to manage the TOE security functions and network access control functions. Security management commands are limited to authorized users (i.e., administrators) only after they have been correctly identified and authenticated. The security management functions are controlled through the use of Admin Roles that can be assigned to TOE users.

1.4.1.2.6 Protection of the TSF

The TOE implements a number of features designed to protect itself to ensure the reliability and integrity of its security features.

It protects particularly sensitive data such as stored passwords and cryptographic keys so that they are not accessible even by an administrator. The TOE uses a clock managed by the OS for reliable time clock information that the TOE uses (e.g., for log accountability).

The TOE uses cryptographic means to protect communication with remote administrators. When the TOE is configured to use the services of a Syslog server or authentication servers in the operational environment, the communication between the TOE and the operational environment component is protected using encryption.

The TOE includes functions to perform self-tests so that it might detect when it is failing. It also includes mechanisms so that the TOE itself can be updated while ensuring that the updates will not introduce malicious or other unexpected changes in the TOE.

1.4.1.2.7 TOE access

The TOE can be configured to display a message of the day banner when an administrator establishes an interactive session and subsequently will enforce an administrator-defined inactivity timeout value after which the inactive session (local or remote) will be terminated.

1.4.1.2.8 Trusted path/channels

The TOE protects interactive communication with administrators using SSHv2 for CLI access. Using SSHv2, both integrity and disclosure protection is ensured.

The TOE protects communication with network peers, such as a log server, and authentication servers (RADIUS and TACACS+) using IPsec connections to prevent unintended disclosure or modification of logs.

1.4.2 TOE Documentation

HPE offers a series of documents that describe the installation process for the TOE as well as guidance for subsequent use and administration of the applicable security features. The following list of documents was examined as part of this evaluation:

1. Preparative Procedures for CC NDPP Evaluated HPE Moonshot-180XGc, 45Gc and 45XGc Switch Module based on Comware V7.1, Version 1.1, 2/9/16
2. Command Reference for CC Supplement, v 1.06, 2/9/2016
3. Comware V7 Configuration Guide for CC Supplement v 1.6, 2/9/16
4. Comware V7 Platform System Log Messages v1.00, 12/2/2015

The following documents for the HPE Moonshot Switch modules can be found under the *General Reference* section of the HPE Moonshot Switch documentation page on the HP Web site. The links for each TOE model are provided below.

- R24xx-HP Moonshot Switch ACL and QoS Command Reference
- R24xx-HP Moonshot Switch Layer 3 - IP Services Command Reference
- R24xx-HP Moonshot Switch Fundamentals Command Reference
- R24xx-HP Moonshot Switch Security Command Reference
- R24xx-HP Moonshot Switch Network Management and Monitoring Command Reference

http://h20566.www2.hpe.com/portal/site/hpsc/public/psi/home/?sp4ts.oid=7398915#manuals
http://h20565.www2.hpe.com/portal/site/hpsc/public/psi/home/?sp4ts.oid=5442834#manuals
http://h20565.www2.hpe.com/portal/site/hpsc/public/psi/home/?sp4ts.oid=8942861#manuals

The following documents for the HPE Moonshot Switch module can be found under the *Setup and Install* section of the HPE Moonshot Switch module documentation page on the HP Web site. The links for each TOE model are provided below.

- R24XX-HP Moonshot Switch ACL and QoS Configuration Guide
- R24XX-HP Moonshot Switch Layer 3 - IP Services Configuration Guide
- R24XX-HP Moonshot Switch Fundamentals Configuration Guide
- R24xx-HP Moonshot Switch Security Configuration Guide
- R24xx-HP Moonshot Switch Network Management and Monitoring Configuration Guide

http://h20566.www2.hpe.com/portal/site/hpsc/public/psi/home/?sp4ts.oid=7398915#manuals

http://h20565.www2.hpe.com/portal/site/hpsc/public/psi/home/?sp4ts.oid=5442834#manuals
http://h20565.www2.hpe.com/portal/site/hpsc/public/psi/home/?sp4ts.oid=8942861#manuals

2. Conformance Claims

This TOE is conformant to the following CC specifications:

- Common Criteria for Information Technology Security Evaluation Part 2: Security functional components, Version 3.1, Revision 4, September 2012.
    - Part 2 Extended
- Common Criteria for Information Technology Security Evaluation Part 3: Security assurance components, Version 3.1 Revision 4, September 2012.
    - Part 3 Conformant
- Package Claims:
    - Protection Profile for Network Devices, Version 1.1 (with Errata #3), 8 June 2012 (NDPP11e3)

2.1 Conformance Rationale

The ST conforms to the NDPP11e3. As explained previously, the security problem definition, security objectives, and security requirements have been drawn from the PP.

3. Security Objectives

The Security Problem Definition may be found in the NDPP11e3 and this section reproduces only the corresponding Security Objectives for operational environment for reader convenience. The NDPP11e3 offers additional information about the identified security objectives, but that has not been reproduced here and the NDPP11e3 should be consulted if there is interest in that material.

In general, the NDPP11e3 has defined Security Objectives appropriate for network infrastructure devices and as such are applicable to the Hewlett-Packard Company Moonshot-180XGc, 45XGc, 45Gc Switch Modules TOE.

3.1 Security Objectives for the Operational Environment

OE.NO_GENERAL_PURPOSE There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE.

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

OE.TRUSTED_ADMIN TOE Administrators are trusted to follow and apply all administrator guidance in a trusted manner.

4. Extended Components Definition

All of the extended requirements in this ST have been drawn from the NDPP11e3. The NDPP11e3 defines the following extended requirements and since they are not redefined in this ST the NDPP11e3 should be consulted for more information in regard to those CC extensions.

Extended SFRs:

- FAU_STG_EXT.1: External Audit Trail Storage
- FCS_CKM_EXT.4: Cryptographic Key Zeroization
- FCS_IPSEC_EXT.1: Explicit: IPSEC
- FCS_RBG_EXT.1: Extended: Cryptographic Operation (Random Bit Generation)
- FCS_SSH_EXT.1: Explicit: SSH
- FIA_PMG_EXT.1: Password Management
- FIA_PSK_EXT.1: Extended: Pre-Shared Key Composition
- FIA_UAU_EXT.2: Extended: Password-based Authentication Mechanism
- FIA_UIA_EXT.1: User Identification and Authentication
- FPT_APW_EXT.1: Extended: Protection of Administrator Passwords
- FPT_SKP_EXT.1: Extended: Protection of TSF Data (for reading of all symmetric keys)
- FPT_TST_EXT.1: TSF Testing
- FPT_TUD_EXT.1: Extended: Trusted Update
- FTA_SSL_EXT.1: TSF-initiated Session Locking

5. Security Requirements

This section defines the Security Functional Requirements (SFRs) and Security Assurance Requirements (SARs) that serve to represent the security functional claims for the Target of Evaluation (TOE) and to scope the evaluation effort.

The SFRs have all been drawn from the NDPP11e3. The refinements and operations already performed in the NDPP11e3 are not identified (e.g., highlighted) here, rather the requirements have been copied from the NDPP11e3 and any residual operations have been completed herein. Of particular note, the NDPP11e3 made a number of refinements and completed some of the SFR operations defined in the Common Criteria (CC) and that PP should be consulted to identify those changes if necessary.

The SARs are also drawn from the NDPP11e3 which includes all the SARs for EAL 1. However, the SARs are effectively refined since requirement-specific 'Assurance Activities' are defined in the NDPP11e3 that serve to ensure corresponding evaluations will yield more practical and consistent assurance than the EAL 1 assurance requirements alone. The NDPP11e3 should be consulted for the assurance activity definitions.

5.1 TOE Security Functional Requirements

The following table identifies the SFRs that are satisfied by Hewlett-Packard Company Moonshot-180XGc, 45XGc, 45Gc Switch Modules TOE.

Requirement Class	Requirement Component
FAU: Security audit	FAU_GEN.1: Audit Data Generation
	FAU_GEN.2: User Identity Association
	FAU_STG_EXT.1: External Audit Trail Storage
FCS: Cryptographic support	FCS_CKM.1: Cryptographic Key Generation (for asymmetric keys)
	FCS_CKM_EXT.4: Cryptographic Key Zeroization
	FCS_COP.1(1): Cryptographic Operation (for data encryption/decryption)
	FCS_COP.1(2): Cryptographic Operation (for cryptographic signature)
	FCS_COP.1(3): Cryptographic Operation (for cryptographic hashing)
	FCS_COP.1(4): Cryptographic Operation (for keyed-hash message authentication)
	FCS_IPSEC_EXT.1: Explicit: IPSEC
	FCS_RBG_EXT.1: Extended: Cryptographic Operation (Random Bit Generation)
	FCS_SSH_EXT.1: Explicit: SSH
FDP: User data protection	FDP_RIP.2: Full Residual Information Protection
FIA: Identification and authentication	FIA_PMG_EXT.1: Password Management
	FIA_PSK_EXT.1: Extended: Pre-Shared Key Composition
	FIA_UAU.7: Protected Authentication Feedback
	FIA_UAU_EXT.2: Extended: Password-based Authentication Mechanism
	FIA_UIA_EXT.1: User Identification and Authentication
FMT: Security management	FMT_MTD.1: Management of TSF Data (for general TSF data)
	FMT_SMF.1: Specification of Management Functions
	FMT_SMR.2: Restrictions on Security Roles
FPT: Protection of the TSF	FPT_APW_EXT.1: Extended: Protection of Administrator Passwords
	FPT_SKP_EXT.1: Extended: Protection of TSF Data (for reading of

	all symmetric keys)
	FPT_STM.1: Reliable Time Stamps
	FPT_TST_EXT.1: TSF Testing
	FPT_TUD_EXT.1: Extended: Trusted Update
FTA: TOE access	FTA_SSL.3: TSF-initiated Termination
	FTA_SSL.4: User-initiated Termination
	FTA_SSL_EXT.1: TSF-initiated Session Locking
	FTA_TAB.1: Default TOE Access Banners
FTP: Trusted path/channels	FTP_ITC.1: Inter-TSF trusted channel
	FTP_TRP.1: Trusted Path

Table 1 TOE Security Functional Components

5.1.1 Security audit (FAU)

5.1.1.1 Audit Data Generation (FAU_GEN.1)

FAU_GEN.1.1

The TSF shall be able to generate an audit record of the following auditable events:
a) Start-up and shut-down of the audit functions;
b) All auditable events for the not specified level of audit; and
c) All administrative actions;
d) Specifically defined auditable events listed in Table 1 (in the NDPP).

Requirement	Auditable Events	Additional Content
FAU_GEN.1	None	None
FAU_GEN.2	None	None
FAU_STG_EXT.1	None	None
FCS_CKM.1	None	None
FCS_CKM_EXT.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_IPSEC_EXT.1	Failure to establish an IPsec SA. Establishment/Termination of an IPsec SA.	Reason for failure. Non-TOE endpoint of connection (IP address) for both successes and failures.
FCS_RBG_EXT.1	None	None
FCS_SSH_EXT.1	Failure to establish an SSH session. Establishment/Termination of an SSH session.	Reason for failure. Non-TOE endpoint of connection (IP address) for both successes and failures.
FDP_RIP.2	None	None
FIA_PMG_EXT.1	None	None
FIA_PSK_EXT.1	None	None
FIA_UAU.7	None	None
FIA_UAU_EXT.2	None	All use of the authentication mechanism.
FIA_UIA_EXT.1	All use of the identification and authentication mechanism.	Provided user identity, origin of the attempt (e.g., IP address).
FMT_MTD.1	None	None
FMT_SMF.1	None	None
FMT_SMR.2	None	None

FPT_APW_EXT.1	None	None
FPT_SKP_EXT.1	None	None
FPT_STM.1	Changes to the time.	None
FPT_TST_EXT.1	None	None
FPT_TUD_EXT.1	Initiation of update.	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_SSL_EXT.1	Any attempts at unlocking 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.

Table 2 Auditable Events

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 1 (in the NDPP).

5.1.1.2 User Identity Association (FAU_GEN.2)

FAU_GEN.2.1

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

5.1.1.3 External Audit Trail Storage (FAU_STG_EXT.1)

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 implementing the [***IPsec***] protocol.

5.1.2 Cryptographic support (FCS)

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

FCS_CKM.1.1

Refinement: The TSF shall generate asymmetric cryptographic keys used for key establishment in accordance with [***-- NIST Special Publication 800-56A, 'Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography' for elliptic curve-based key establishment schemes and implementing 'NIST curves' P-256, P-384 and [P-521] (as defined in FIPS PUB 186-3, 'Digital Signature Standard');***

- NIST Special Publication 800-56B, 'Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography' for RSA-based key establishment schemes] and specified cryptographic key sizes equivalent to, or greater than, a symmetric key strength of 112 bits.

5.1.2.2 Cryptographic Key Zeroization (FCS_CKM_EXT.4)

FCS_CKM_EXT.4.1

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

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

FCS_COP.1(1).1

Refinement: The TSF shall perform encryption and decryption in accordance with a specified cryptographic algorithm AES operating in [***CBC, GCM***] and cryptographic key sizes 128-bits and 256-bits that meets the following:
- FIPS PUB 197, 'Advanced Encryption Standard (AES)'
- [***NIST SP 800-38A***].

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

FCS_COP.1(2).1

Refinement: The TSF shall perform cryptographic signature services in accordance with a [***(2) RSA Digital Signature Algorithm (rDSA) with a key size (modulus) of 2048 bits or greater***] that meets the following:
[***Case: Elliptic Curve Digital Signature Algorithm - FIPS PUB 186-3, 'Digital Signature Standard' - The TSF shall implement 'NIST curves' P-256, P-384 and [P-521] (as defined in FIPS PUB 186-3, 'Digital Signature Standard')***].

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

FCS_COP.1(3).1

Refinement: The TSF shall perform cryptographic hashing services in accordance with a specified cryptographic algorithm [***SHA-1, SHA-256, SHA-384, SHA-512***] and message digest sizes [***160, 256, 384, 512***] bits that meet the following: FIPS Pub 180-3, 'Secure Hash Standard.'

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

FCS_COP.1(4).1

Refinement: The TSF shall perform keyed-hash message authentication in accordance with a specified cryptographic algorithm HMAC-[***SHA-1, SHA-256, SHA-384, SHA-512***], key size [***160bits,256bits, 384bits, 512bits***], and message digest sizes [***160, 256, 384, 512***] bits that meet the following: FIPS Pub 198-1, 'The Keyed-Hash Message Authentication Code', and FIPS Pub 180-3, 'Secure Hash Standard.'

5.1.2.7 Explicit: IPSEC (FCS_IPSEC_EXT.1)

FCS_IPSEC_EXT.1.1

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

FCS_IPSEC_EXT.1.2

The TSF shall implement [***tunnel mode, transport mode***]

FCS_IPSEC_EXT.1.3

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

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 (as specified by RFC 3602) together with a Secure Hash Algorithm (SHA)-based HMAC, AES-CBC-256 (as specified by RFC 3602) together with a Secure Hash Algorithm (SHA)-based HMAC, AES-GCM-128 as specified in RFC 4106, AES-GCM-256 as specified in RFC 4106,***].

FCS_IPSEC_EXT.1.5
The TSF shall implement the protocol: [***IKEv2 as defined in RFCs 5996 (with mandatory support for NAT traversal as specified in section 2.23), 4307, and [RFC 4868 for hash functions]***].

FCS_IPSEC_EXT.1.6
The TSF shall ensure the encrypted payload in the [***IKEv2***] protocol uses the cryptographic algorithms AES-CBC-128, AES-CBC-256 as specified in RFC 6379 and [***no other algorithm***].

FCS_IPSEC_EXT.1.7
The TSF shall ensure that IKEv1 Phase 1 exchanges use only main mode

FCS_IPSEC_EXT.1.8
The TSF shall ensure that [***IKEv2 SA lifetimes can be established based on [number of packets/number of bytes; length of time, where the time values can be limited to: 24 hours for Phase 1[1] SAs and 8 hours for Phase 2[2] SAs***].

FCS_IPSEC_EXT.1.9
The TSF shall ensure that all IKE protocols implement DH Groups 14 (2048-bit MODP), and **[*24 (2048-bit MODP with 256-bit POS), 19 (256-bit Random ECP),* and *20 (384-bit Random ECP*]**.

FCS_IPSEC_EXT.1.10
The TSF shall ensure that all IKE protocols implement Peer Authentication using the [***RSA, ECDSA***] algorithm and Pre-shared Keys.

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

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

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

5.1.2.9 Explicit: SSH (FCS_SSH_EXT.1)

FCS_SSH_EXT.1.1
The TSF shall implement the SSH protocol that complies with RFCs 4251, 4252, 4253, 4254, and [***5656***].

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

FCS_SSH_EXT.1.4
The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms: AES-CBC-128, AES-CBC-256, [***AEAD_AES_128_GCM, AEAD_AES_256_GCM***].

FCS_SSH_EXT.1.5
The TSF shall ensure that the SSH transport implementation uses [***SSH_RSA, ecdsa-sha2-nistp256***] and [***ecdsa-sha2-nistp384***] as its public key algorithm(s).

FCS_SSH_EXT.1.6
The TSF shall ensure that data integrity algorithms used in SSH transport connection is [***HMAC-SHA1, HMAC-SHA1-96***].

FCS_SSH_EXT.1.7
The TSF shall ensure that diffie-hellman-group14-sha1 and [***ecdh-sha2-nistp256, ecdh-sha2-nistp384***] are the only allowed key exchange methods used for the SSH protocol.

[1] That is, IKE_SA_INIT and IKE_AUTH exchanges in IKEv2.
[2] That is, CREATE_CHILD_SA exchange in IKEv2.

5.1.3 User data protection (FDP)

5.1.3.1 Full Residual Information Protection (FDP_RIP.2)

FDP_RIP.2.1

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

5.1.4 Identification and authentication (FIA)

5.1.4.1 Password Management (FIA_PMG_EXT.1)

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 special characters: : [***"!", "@", "#", "$", "%", "^", "&", "*", "(", ")", [" ' ", "+", ",", "-", ".", "/", ":", ";", "<", "=", ">", "[", "\", "]", "_ ", "`", "{", "}", and "~"]***];
2. Minimum password length shall settable by the Security Administrator, and support passwords of 15 characters or greater.

5.1.4.2 Extended: Pre-Shared Key Composition (FIA_PSK_EXT.1)

FIA_PSK_EXT.1.1

The shall be able to use pre-shared keys for IPsec.

FIA_PSK_EXT.1.2

The TSF shall be able to accept text-based pre-shared keys that:

- are 22 characters and [***[lengths from 15 to 128 characters]***];
- composed of any combination of upper and lower case letters, numbers, and special characters (that include: '!', '@', '#', '$', '%', '^', '&', '*', '(', and ')').

FIA_PSK_EXT.1.3

The TSF shall [***condition the text-based pre-shared keys by using [[the bit representation of the ASCII coding of the entered characters as the key] and use no other pre-shared keys]***].

5.1.4.3 Protected Authentication Feedback (FIA_UAU.7)

FIA_UAU.7.1

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

5.1.4.4 Extended: Password-based Authentication Mechanism (FIA_UAU_EXT.2)

FIA_UAU_EXT.2.1

The TSF shall provide a local password-based authentication mechanism, [***[and access to external RADIUS and TACACS+]***] to perform administrative user authentication.

5.1.4.5 User Identification and Authentication (FIA_UIA_EXT.1)

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;
- [***[network switching services]***].

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.

5.1.5 Security management (FMT)

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

FMT_MTD.1.1

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

5.1.5.2 Specification of Management Functions (FMT_SMF.1)

FMT_SMF.1.1

The TSF shall be capable of performing the following management functions:
- Ability to administer the TOE locally and remotely;
- Ability to update the TOE, and to verify the updates using [***digital signature***] capability prior to installing those updates;
- [***Ability to configure the cryptographic functionality***].

5.1.5.3 Restrictions on Security Roles (FMT_SMR.2)

FMT_SMR.2.1

The TSF shall maintain the roles: Authorized Administrator.

FMT_SMR.2.2

The TSF shall be able to associate users with roles.

FMT_SMR.2.3

The TSF shall ensure that the conditions
- Authorized Administrator role shall be able to administer the TOE locally;
- Authorized Administrator role shall be able to administer the TOE remotely;

are satisfied

5.1.6 Protection of the TSF (FPT)

5.1.6.1 Extended: Protection of Administrator Passwords (FPT_APW_EXT.1)

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.

5.1.6.2 Extended: Protection of TSF Data (for reading of all symmetric keys) (FPT_SKP_EXT.1)

FPT_SKP_EXT.1.1

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

5.1.6.3 Reliable Time Stamps (FPT_STM.1)

FPT_STM.1.1

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

5.1.6.4 TSF Testing (FPT_TST_EXT.1)

FPT_TST_EXT.1.1

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

5.1.6.5 Extended: Trusted Update (FPT_TUD_EXT.1)

FPT_TUD_EXT.1.1

The TSF shall provide security administrators the ability to query the current version of the TOE firmware/software.

FPT_TUD_EXT.1.2

The TSF shall provide security administrators the ability to initiate updates to TOE firmware/software.

FPT_TUD_EXT.1.3

The TSF shall provide a means to verify firmware/software updates to the TOE using a [***digital signature mechanism***] prior to installing those updates.

5.1.7 TOE access (FTA)

5.1.7.1 TSF-initiated Termination (FTA_SSL.3)

FTA_SSL.3.1

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

5.1.7.2 User-initiated Termination (FTA_SSL.4)

FTA_SSL.4.1

The TSF shall allow Administrator-initiated termination of the Administrator's own interactive session.

5.1.7.3 TSF-initiated Session Locking (FTA_SSL_EXT.1)

FTA_SSL_EXT.1.1

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

5.1.7.4 Default TOE Access Banners (FTA_TAB.1)

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.

5.1.8 Trusted path/channels (FTP)

5.1.8.1 Inter-TSF trusted channel (FTP_ITC.1)

FTP_ITC.1.1

Refinement: The TSF shall use [***IPsec***] 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 [***transmitting audit records to an audit server, and external authentication functions***].

5.1.8.2 Trusted Path (FTP_TRP.1)

FTP_TRP.1.1

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

FTP_TRP.1.2

Refinement: 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.

5.2 TOE Security Assurance Requirements

The SARs for the TOE are the components as specified in Part 3 of the Common Criteria. Note that the SARs have effectively been refined with the assurance activities explicitly defined in association with both the SFRs and SARs.

Requirement Class	Requirement Component
ADV: Development	ADV_FSP.1: Basic functional specification
AGD: Guidance documents	AGD_OPE.1: Operational user guidance
	AGD_PRE.1: Preparative procedures
ALC: Life-cycle support	ALC_CMC.1: Labelling of the TOE
	ALC_CMS.1: TOE CM coverage
ATE: Tests	ATE_IND.1: Independent testing - conformance
AVA: Vulnerability assessment	AVA_VAN.1: Vulnerability survey

Table 3 Assurance Components

5.2.1 Development (ADV)

5.2.1.1 Basic functional specification (ADV_FSP.1)

ADV_FSP.1.1d

The developer shall provide a functional specification.

ADV_FSP.1.2d

The developer shall provide a tracing from the functional specification to the SFRs.

ADV_FSP.1.1c

The functional specification shall describe the purpose and method of use for each SFR-enforcing and SFR-supporting TSFI.

ADV_FSP.1.2c

The functional specification shall identify all parameters associated with each SFR-enforcing and SFR-supporting TSFI.

ADV_FSP.1.3c

The functional specification shall provide rationale for the implicit categorisation of interfaces as SFR-non-interfering.

ADV_FSP.1.4c

The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification.

ADV_FSP.1.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_FSP.1.2e

The evaluator shall determine that the functional specification is an accurate and complete instantiation of the SFRs.

5.2.2 Guidance documents (AGD)

5.2.2.1 Operational user guidance (AGD_OPE.1)

AGD_OPE.1.1d

The developer shall provide operational user guidance.

AGD_OPE.1.1c

The operational user guidance shall describe, for each user role, the user-accessible functions and privileges that should be controlled in a secure processing environment, including appropriate warnings.

AGD_OPE.1.2c

The operational user guidance shall describe, for each user role, how to use the available interfaces provided by the TOE in a secure manner.

AGD_OPE.1.3c

The operational user guidance shall describe, for each user role, the available functions and interfaces, in particular all security parameters under the control of the user, indicating secure values as appropriate.

AGD_OPE.1.4c

The operational user guidance shall, for each user role, clearly present each type of security-relevant event relative to the user-accessible functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.

AGD_OPE.1.5c

The operational user guidance shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation.

AGD_OPE.1.6c

The operational user guidance shall, for each user role, describe the security measures to be followed in order to fulfil the security objectives for the operational environment as described in the ST.

AGD_OPE.1.7c

The operational user guidance shall be clear and reasonable.

AGD_OPE.1.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.2.2.2 Preparative procedures (AGD_PRE.1)

AGD_PRE.1.1d

The developer shall provide the TOE including its preparative procedures.

AGD_PRE.1.1c

The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.

AGD_PRE.1.2c

The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the operational environment in accordance with the security objectives for the operational environment as described in the ST.

AGD_PRE.1.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AGD_PRE.1.2e

The evaluator shall apply the preparative procedures to confirm that the TOE can be prepared securely for operation.

5.2.3 Life-cycle support (ALC)

5.2.3.1 Labelling of the TOE (ALC_CMC.1)

ALC_CMC.1.1d

The developer shall provide the TOE and a reference for the TOE.

ALC_CMC.1.1c

The TOE shall be labelled with its unique reference.

ALC_CMC.1.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.2.3.2 TOE CM coverage (ALC_CMS.1)

ALC_CMS.1.1d

The developer shall provide a configuration list for the TOE.

ALC_CMS.1.1c

The configuration list shall include the following: the TOE itself; and the evaluation evidence required by the SARs.

ALC_CMS.1.2c

The configuration list shall uniquely identify the configuration items.

ALC_CMS.1.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.2.4 Tests (ATE)

5.2.4.1 Independent testing - conformance (ATE_IND.1)

ATE_IND.1.1d

The developer shall provide the TOE for testing.

ATE_IND.1.1c

The TOE shall be suitable for testing.

ATE_IND.1.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ATE_IND.1.2e

The evaluator shall test a subset of the TSF to confirm that the TSF operates as specified.

5.2.5 Vulnerability assessment (AVA)

5.2.5.1 Vulnerability survey (AVA_VAN.1)

AVA_VAN.1.1d

The developer shall provide the TOE for testing.

AVA_VAN.1.1c

The TOE shall be suitable for testing.

AVA_VAN.1.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_VAN.1.2e

The evaluator shall perform a search of public domain sources to identify potential vulnerabilities in the TOE.

AVA_VAN.1.3e

The evaluator shall conduct penetration testing, based on the identified potential vulnerabilities, to determine that the TOE is resistant to attacks performed by an attacker possessing Basic attack potential.

6. TOE Summary Specification

This chapter describes the security functions:

- Security audit
- Cryptographic support
- User data protection
- Identification and authentication
- Security management
- Protection of the TSF
- TOE access
- Trusted path/channels

6.1 Security audit

The TOE is designed to produce syslog conformant messages in a number security relevant events (the success and failure login of the user, regardless of the authentication mechanism; changing a user's password; and adding and deleting user accounts). In each case the audit record includes the time and date, identification of the responsible subject (e.g., by network address or user ID), the type of event, the outcome of the event, and other information depending on the event type. The TOE generates audit records for all events listed in **Table 2 Auditable Events** as well as start-up and shutdown of audit and all administrative actions.

The TOE includes an internal log implementation that can be used to store and review audit records locally. The maximum storage space reserved for the local log file can be configured to a range between 1 and 10MB. When the local log storage is full, the TOE will overwrite the oldest records with new records. Only users with the role network-admin, network-operator, or level-15 can access the local audit trail. Alternately, the TOE can be configured to send generated audit records to an external Syslog server using IPsec.

Note that audit records are not buffered for transmission to the syslog server. If the connection to the syslog server goes down, generated audit records are not queued and will not be transmitted to the syslog server when the connection is re-established. However, audit records will still be delivered to any other configured audit destinations, such as the log buffer and local log file. Additionally, the TOE generates audit records when connection to the syslog server is lost and when it is restored, and these audit records are sent to any other configured audit destinations. Therefore, the administrator is advised to ensure additional audit destinations are configured so that generated audit records will still be available for review in the event of loss of connectivity to the syslog server. In addition, multiple log servers can be configured to provide redundancy.

The Security audit function is designed to satisfy the following security functional requirements:

- FAU_GEN.1: The TOE generates audit events for the not specified level of audit. A syslog server in the environment is relied on to store audit records generated by the TOE.
- FAU_GEN.2: The TOE identifies the responsible user for each event based on the specific administrator or network entity (identified by IP address) that caused the event.
- FAU_STG_EXT.1: The TOE can be configured to export audit records to an external SYSLOG server. This communication is protected with the use of IPsec.

6.2 Cryptographic support

The TOE includes a crypto module providing supporting cryptographic functions. The evaluated configuration requires that the TOE be configured in Common Criteria mode to ensure CAVP tested functions are used.

The following functions have been CAVP tested in accordance with the identified standards:

Functions	Standards	Certificates
Asymmetric key generation		
• ECC key pair generation (NIST curves P-256, P-384 and P-521)	NIST Special Publication 800-56A	738
• Domain parameter generation (key size 2048 bits)	NIST Special Publication 800-56B	1969
Encryption/Decryption		
• AES CBC, CTR, and GCM (128, 256 bits)	FIPS PUB 197 NIST SP 800-38A NIST SP 800-38D	3855
Cryptographic signature services		
• RSA Digital Signature Algorithm (rDSA) (modulus 2048) • ECDSA (NIST curves P-256, P-384 and P-521)	FIPS PUB 186-2 FIPS PUB 186-3	1969 834
Cryptographic hashing		
• SHA-1, SHA-256, SHA-384 and SHA-512 (digest sizes 160, 224, 256, 384 and 512 bits)	FIPS Pub 180-3	3177
Keyed-hash message authentication		
• HMAC-SHA-1 (block size 512 bits, key size 160 bits and digest size 160 bits)	FIPS Pub 198-1 FIPS Pub 180-3	2503
• HMAC-SHA-256 (block size 512 bits, key Size 256 bits and digest size 256 bits) • HMAC-SHA-384 (block size 1024 bits, key Size 384 bits and digest size 384 bits) • HMAC-SHA-512 (block size 1024 bits, key Size 512 bits and digest size 512 bits)	FIPS Pub 198-1 FIPS Pub 180-3	2503
Random bit generation		
• CTR_DRBG (AES) with one independent software-based noise source of 256 of non-determinism	NIST Special Publication 800-90	1094

Table 4 Cryptographic Functions

The TOE is designed to zeroize secret and private cryptographic keys and critical security parameters (CSPs) when they are no longer required by the TOE. The following table identifies the applicable secret and private cryptographic keys and CSPs, and summarizes how and when they are deleted. Note that only some of the keys and CSPs are applicable to the evaluation. Also note that where identified zeroization occurs as follows: 1) when deleted from FLASH, the previous value is overwritten once with zeroes; 2) when added or changed in FLASH, any old value is overwritten completely with the new value; and, 3) the zeroization of values in RAM is achieved by overwriting once with zeroes.

#	Key/ CSP Name	Generation/ Algorithm	Key Size	Description	Storage	Zeroization
Public key management						

#	Key/ CSP Name	Generation/ Algorithm	Key Size	Description	Storage	Zeroization
CSP1-1	RSA private key	CTR_DRBG (AES)/RSA	2048 bits	Identity certificates for the security appliance itself and also used in IPsec and SSH negotiations.	FLASH (cipher text / AES-CTR 256)	Using CLI command "**public-key local destroy rsa** …" to zeroize.
CSP1-2	DSA private key *(note that DSA is not included in the evaluated configuration)*	CTR_DRBG (AES)/DSA	2048 bits	Identity certificates for the security appliance itself and also used in SSH negotiations.	FLASH (cipher text /AES-CTR 256)	Using CLI command "**public-key local destroy dsa** …" to zeroize
CSP1-3	ECDSA private key	CTR_DRBG(AES)/ECDSA	NIST P256, P384, P521	Identity certificates for the security appliance itself and also used in IPsec, SSH and SSL.	FLASH (cipher text /AES-CTR 256)	Using CLI command "public-key local destroy ecdsa …" to zeroize.
CSP1-4	RSA Public keys	RSA	RSA:1024 ~ 2048 bits Note: 192 – bit keys are not used in the evaluated configuration	Public keys of peers to validate the digital signature	FLASH(plain text)	Peer public keys exist in a FLASH start-up configuration file. Using CLI commands "**undo public-key peer** " and "**save**" to zeroize the public keys.
CSP1-5	DSA Public keys *(note that DSA is not included in the evaluated configuration)*	DSA	1024 ~ 2048 bits	Public keys of peers to validate the digital signature	FLASH(plain text)	Peer public keys exist in a FLASH start-up configuration file. Using CLI commands "**undo public-key peer** " and "**save**" to zeroize the public keys.
CSP1-6	ECDSA Public keys	ECDSA	NIST P256, P384, P521	Public keys of peers to validate the digital signature	FLASH (plain text)	Peer public keys exist in a FLASH start-up configuration file. Using CLI commands "**undo public-key peer** " and "**save**" to zeroize the public keys.
IPsec						
CSP2-1	IPsec authentication keys	Generated using IKE protocol (CTR_DRBG (AES)+HMAC-SHA1/HMAC-SHA256+DH). Algorithms: HMAC-SHA1-96 HMAC-SHA-256-128 HMAC-SHA-384-192	160 bits 256 bits 384 bits 512 bits AES-GMAC: 128, 256 bits Note: GMAC is not used in any of the evaluated mechanisms	Used for authenticating the IPsec traffic	RAM (plain text)	Zeroized upon deleting the IPsec session.

#	Key/ CSP Name	Generation/ Algorithm	Key Size	Description	Storage	Zeroization
		HMAC-SHA-512-256 AES-GMAC				
CSP2-2	IPsec encryption keys	Generated using IKE protocol (CTR_DRBG (AES)+HMAC-SHA1/HMAC-SHA256+DH). Algorithms: AES-CBC, AES-GCM	128 bits 192 bits 256 bits Note: 192 – bit keys are not used in the evaluated configuration	Used for encrypting the IPsec traffic	RAM (plain text)	Zeroized upon deleting the IPsec session.
CSP2-3	IPsec authentication keys	HMAC-SHA1-96 HMAC-SHA-256-128 HMAC-SHA-384-192 HMAC-SHA-512-256	160 bits 256 bits 384 bits 512 bits	Manually configured key used for authenticating the IPsec traffic.	FLASH (cipher text / AES-CTR 256) and RAM (plain text)	Keys will be zeroized using CLI commands "**undo sa hex-key authentication …**" and " **save**",
CSP2-4	IPsec encryption keys	AES	128 bits 192 bits 256 bits Note: 192 – bit keys are not used in the evaluated configuration	Manually configured key used for encrypting the IPsec traffic.	FLASH (cipher text / AES-CTR 256) and RAM (plain text)	Keys will be zeroized using CLI commands "**undo sa hex-key encryption …**" and " **save**",
IKEv1						
CSP3-1	IKE pre-shared keys	Shared Secret	15 ~ 128 bytes	Entered by the Crypto-Officer in plain text form and used for authentication during IKE	FLASH (cipher text/ AES-CTR 256) and RAM (plain)	Keys will be zeroized using CLI commands "**undo pre-shared-key …**" and " **save**",
CSP3-2	IKE RSA Authentication private Key	RSA DSA	RSA: 2048 bits DSA: 2048 bits	private key used for IKE protocol during the handshake	RAM(plain text)	Automatically zeroized upon handshake finishing
CSP3-3	IKE Diffie-Hellman Key Pairs	CTR_DRBG (AES) / DH	2048 bits	Key agreement for IKE	RAM (plain text)	Automatically zeroized upon handshake finishing
CSP3-4	IKE Integrity key	Generated using IKE (CTR_DRBG (AES)+HMAC-SHA1/HMAC-SHA256+DH). Algorithms: HMAC-SHA1, HMAC-SHA256	160 bits 256 bits	Used for integrity test of IKE negotiations	RAM (plain text)	Zeroized upon deleting the IKE session.

#	Key/ CSP Name	Generation/ Algorithm	Key Size	Description	Storage	Zeroization
CSP3-5	IKE Encryption Key	Generated using IKE (CTR_DRBG (AES)+HMAC-SHA1/HMAC-SHA256+DH). Algorithms: AES	128 bits, 192 bits, 256 bits Note: 192 – bit keys are not used in the evaluated configuration	Used for encrypting IKE negotiations	RAM (plain text)	Zeroized upon deleting the IKE session.
IKEv2						
CSP4-1	IKE pre-shared keys	Shared Secret	15 ~ 128 bytes	Entered by the Crypto-Officer in plain text form and used for authentication during IKE	FLASH(cipher text/ AES-CTR 256) and RAM (plain)	Keys will be zeroized using CLI commands "**undo pre-shared-key …**" and " **save**",
CSP4-2	IKE RSA Authentication private Key	RSA DSA ECDSA	RSA:2048 bits DSA:2048 bits ECDSA:P-256, P-384	private key used for IKE protocol during the handshake	RAM(plain text)	Automatically zeroized upon handshake finishing
CSP4-3	IKE Diffie-Hellman Key Pairs	CTR_DRBG (AES) / DH,ECDH	DH:2048 bits ECDH:P-256, P-384	Key agreement for IKE	RAM (plain text)	Automatically zeroized upon handshake finishing
CSP4-4	IKE Integrity key	Generated using IKE (CTR_DRBG (AES)+DH/ECDH + HMAC-SHA1/HMAC-SHA256/HMAC-SHA384). Algorithms: HMAC-SHA1, HMAC-SHA256-128 HMAC-SHA384-192	160 bits 256 bits 384 bits	Used for integrity test of IKE negotiations	RAM (plain text)	Zeroized upon deleting the IKE session.
CSP4-5	IKE Encryption Key	Generated using IKE (CTR_DRBG (AES)+DH/ECDH + HMAC-SHA1/HMAC-SHA256/HMAC-SHA384). Algorithms: AES	128 bits, 192 bits, 256 bits Note: 192 – bit keys are not used in the evaluated configuration	Used for encrypting IKE negotiations	RAM (plain text)	Zeroized upon deleting the IKE session.
SSH						
CSP5-1	SSH Private key	RSA ECDSA	RSA:2048 bits ECDSA: P-256, P-384	private key used for SSH protocol during handshake	RAM(plain text)	Automatically zeroized upon finishing handshake.

#	Key/ CSP Name	Generation/ Algorithm	Key Size	Description	Storage	Zeroization
CSP5-2	SSH Diffie-Hellman Key Pairs	CTR_DRBG (AES) / DH/ECDH	DH: 2048 bits ECDH: P-256, P-384	Key agreement for SSH sessions.	RAM (plain text)	Automatically zeroized upon finishing handshake.
CSP5-3	SSH Session encryption key	Generated using the SSH protocol(CTR_DRBG(AES)+SHA1+DH) Algorithms: AES-CBC, AES-GCM	128 bits, 256 bits	Key used for encrypting SSH session.	RAM (plain text)	Automatically zeroized when SSH session terminated.
CSP5-4	SSH Session authentication key	Generated using the SSH protocol(CTR_DRBG(AES)+SHA1+DH) Algorithms: HMAC-SHA1, HMAC-SHA1-96 AES-GCM	SHA1: 160 bits AES-GCM: 128 bits, 256 bits	Key used for authenticating SSH session.	RAM (plain text)	Automatically zeroized when SSH session terminated.
AAA						
CSP6-1	User Passwords	Secret	15 ~ 63 bytes	Critical security parameters used to authenticate the administrator login.	FLASH (hashed text/SHA-512) and RAM (plain)	Use CLI command "**password**" to set new password, or use CLI command "**undo local-user** …" to zeroize the password and delete user account.
CSP6-2	Super password	Secret	15 ~ 63 bytes	Critical security parameters used to authenticate privilege promoting.	FLASH (hashed text/SHA-512) and RAM (plain)	Use CLI command "**undo super password**" to zeroize the super password.
CSP6-3	RADIUS shared secret keys	Shared Secret	15 ~ 64 bytes	Used for authenticating the RADIUS server to the security appliance and vice versa. Entered by the Security administrator in plain text form and stored in cipher text form.	FLASH (cipher text/ AES-CTR 256) and RAM (plain)	Keys will be zeroized using following commands: "**undo primary authentication**", ""**undo primary accounting",** "**undo secondary authentication**", ""**undo secondary accounting".**
CSP6-4	TACACS+ shared secret keys	Shared Secret	15~255 bytes	Used for authenticating the TACACS+ server to the security appliance and vice versa. Entered by the Security administrator in plain text form and stored in cipher text form.	FLASH (cipher text/ AES-CTR 256) and RAM (plain)	Keys will be zeroized using following commands: "**undo primary authentication**", ""**undo primary accounting",** ""**undo primary authorization",**

#	Key/ CSP Name	Generation/ Algorithm	Key Size	Description	Storage	Zeroization
						"**undo secondary authentication**", ""**undo secondary accounting**", ""**undo secondary authorization**".
Random Bits Generation						
CSP7-1	DRBG seed	Entropy / SP 800 - 90 CTR_DRBG	256 bits	Input to the DRBG that determines the internal state of the DRBG	RAM (plaintext)	Automatically zeroized when DRBG initialized
CSP7-2	DRBG V	SP 800 - 90 CTR_DRBG	128 bits	Generated by entropy source via the CTR_DRBG derivation function	RAM (plaintext)	Resetting or rebooting the security appliance
CSP7-3	DRBG Key	SP 800 - 90 CTR_DRBG	256 bits	Generated by entropy source via the CTR_DRBG derivation function	RAM (plaintext)	Resetting or rebooting the security appliance

Table 5 Key/CSP Zeroization Summary

These supporting cryptographic functions are included to support the SSHv2 (RFCs 4251, 4252, 4253, and 4254) secure communication protocol.

The TOE supports SSHv2 with AES (CBC, GCM) 128 or 256 bit ciphers, in conjunction with HMAC-SHA-1 or HMAC-SHA-1-96. The TOE supports public key algorithms RSA, ecdsa-sha2-nistp256, and ecdsa-sha2-nistp384. It supports diffie-hellman-group14-sha1, ecdh-sha2-nistp256 and with ecdh-sha2-nistp384 key exchange methods. While DES and 3DES (CBC), HMAC-MD5 and HMAC-MD5-96, as well as diffie-hellman-group-1 and diffie-hellman-exchange are all implemented, they are disabled while the TOE is operating in CC/FIPS mode.

SSHv2 connections are rekeyed prior to reaching 228 packets. The authentication timeout period is 90 seconds allowing clients to retry only 3 times. Both public-key and password based authentication can be configured. Packets are limited to 256K bytes. Note that the TOE manages a packet counter for each SSH session so that it can initiate a new key exchange when the 228 packet limit is reached. Whenever the timeout period or authentication retry limit is reached, the TOE closes the applicable TCP connection and releases the SSH session resources. As SSH packets are being received, the TOE uses a buffer to build all packet information. Once complete, the packet is checked to ensure it can be appropriately decrypted. However, if it is not complete when the buffer becomes full (256K bytes) the packet will be dropped.

The TOE includes an implementation of IPsec in accordance with RFC 4301. The primary cryptographic algorithms used by the TOE include AES-GCM-128, AES-GCM-256, AES-CBC-128 and AES-CBC-256 (specified by RFCs 4106 and 3602) along with IKEv2 as defined in RFCs 5996 (with mandatory support for NAT traversal as specified in section 2.23), 4307, and 4868 for hash functions. Table 5 Key/CSP Zeroization Summary sections IPsec and

IKEv2 identify HMAC support for key generation, authentication, and integrity. The TOE supports the 128-bit and 256-bit AES for both IKE_SA and CHILD_SAs. The TOE supports both tunnel and transport modes.

The TOE provides mechanisms to implement an IPsec Security Policy Database (SPD) and to process packets to satisfy the behavior of DISCARD, BYPASS and PROTECT packet processing as described in RFC 4301. This is achieved through the administrator configuring appropriately specified access control lists (ACLs). The administrator first establishes an IPsec Policy containing a Security ACL to match traffic to be encrypted (PROTECTed) and applies it to the outbound interface. The Security ACL contains one or more rules, which are ordered based on a numeric index from lowest to highest. The TOE compares packets in turn against each rule in the Security ACL to determine if the packet matches the rule. Packets can be matched based on protocol (for example, TCP, UDP), source IP address and destination IP address. As soon as a match is found, the packet is handled based on the action specified in the rule—either permit, which equates to PROTECT, or deny, which equates to BYPASS. Traffic matching a deny rule or not matching any rule in the Security ACL is passed on to the next stage of processing. Note that multiple IPsec Policies can be assigned to an interface as a policy group. In this case, each policy in the group has its own priority number that is unique within the policy group. Each policy is considered in turn, starting at the lowest number policy (which has highest priority) and proceeding in turn with increasing policy numbers until a match is found or until all policies have been examined. To cater for packets that match a deny rule or do not match any of the IPsec Policies, the administrator needs to configure further ACLs and bind them to the outbound interface using the packet-filter command. These ACLs specify permit/deny rules to implement BYPASS/DISCARD behavior. As with the Security ACL, the TOE compares packets against rules in the packet filtering ACL based on protocol, source IP address and destination IP address. The rules in the packet filtering ACL can be ordered in the same fashion as in a Security ACL. In the packet filtering ACL, a permit rule equates to BYPASS, and a deny rule equates to DISCARD. By default, the packet filter permits packets that do not match any ACL rule. In the evaluated configuration, an administrator changes this action to deny.

IKEv2 SA lifetime and volume limits can be configured by an authorized administrator. IKE_SA lifetime can be limited to 24 hours (actually any value between 120 and 86,400 seconds). CHILD_SA lifetimes can be limited to 8 hours (actually any value from 180 to 604,800 seconds). Volume can be limited to as little as 2.5 MB (actually any value between 2,560 and 4,294,967,295 KB).

The IKEv2 protocol implemented by the TOE includes DH 14 (2048-bit MODP), 24 (2048-bit MODP with 256-bit POS), 19 (256-bit Random ECP), and 20 (384-bit Random ECP) using RSA and ECDSA peer authentication. In the IKE_SA_INIT and CREATE_CHILD_SA exchanges, the TOE and peer will agree on the DH group both can support. When the TOE initiates IKE negotiation, the DH group is sent in order according to the peer's configuration. When the TOE receives an IKE proposal, it will select the first match and the negotiation will fail if there is no match.

The TOE can be configured to use pre-shared keys with a given peer. When a pre-shared key is configured, the IPsec tunnel will be established using the configured pre-shared key, provided that the peer also has the pre-shared key. Text-based pre-shared keys used for IPsec can be constructed of essentially any alphabetic character (upper and lower case), numerals, and special characters (for example, "!", "@", "#", "$", "%", "^", "&", "*", "(", and ")") and can be anywhere from 15 to 128 characters in length (including, for example, 22 characters). In this case, the TOE uses the bit representation of the underlying ASCII characters of the text-based pre-shared key as the key for IPsec peer authentication. The TOE requires suitable keys to be entered by an authorized administrator.

The Cryptographic support function is designed to satisfy the following security functional requirements:

- FCS_CKM.1: See table above
- FCS_CKM_EXT.4: See table above
- FCS_COP.1(1): See table above
- FCS_COP.1(2): See table above
- FCS_COP.1(3): See table above
- FCS_COP.1(4): See table above

- FCS_IPSEC_EXT.1: The TOE supports IPsec as indicated above to protect when exporting audit records and when communicating with authentication server.
- FCS_RBG_EXT.1: See table above
- FCS_SSH_EXT.1: The TOE supports SSHv2 command-line secure administrator sessions as indicated above
- FIA_PSK_EXT.1: The TOE supports pre-shared keys for IPsec peer authentication.

6.3 User data protection

The TOE is designed to ensure its own internal integrity as well as to protect user data from potential, unintended reuse by clearing resources (e.g., memory) as they are allocated to create objects used in the implementation of the TOE operations. Note that volatile memory is the primary resource involved in normal TOE execution while its persistent storage is based on non-volatile flash memory.

When a network packet is sent, the buffer used by the packet is recalled and managed by the buffer pool. After that, if a new packet acquires a buffer from the buffer pool, the new packet data will be used to overwrite any previous data in the buffer. If an allocated buffer exceeds the size of the packet, the additional space will be overwritten (padded) with zeros.

The User data protection function is designed to satisfy the following security functional requirements:

- FDP_RIP.2: The TOE always overwrites resources when allocated for use in objects.

6.4 Identification and authentication

The TOE defines administrative users in terms of:

- User identity,
- User name,
- Password, and
- Role.

Specific roles are associated with users and serve to determine the functions the associated user can perform.

The TOE authenticates administrative users connecting to the TOE CLI via a local console or remotely using SSHv2 in the same manner using either its own password-based authentication mechanism or its internal RADIUS or TACACS+ servers. In order for an administrative user to access the TOE (i.e., to perform any functions except to see a configure login banner or to access network access control services, including processing RADIUS, TACACS+, and other authentication requests from external entities), an administrative user account must be created for the user with an assigned role.

The TOE password authentication mechanism enforces password composition rules. A minimum password length can be configured to 15 characters. Passwords can generally contain alphabetic (upper or lower case) characters, numeric characters, and special characters such as any of "!@#$%^&*() {}?_=.,+-<>/" and they are case-sensitive

The TOE supports both public key-based and password-based client authentication for the SSH trusted path. To successfully establish an interactive administrative session, the authorized remote administrator must provide either the correct public key or both a password and the correct public key for successful authentication.

When configuring IPsec connections, both certificate- and pre-shared-key based authentication are supported. In the case of pre-shared keys, the administrator types in and confirms the pre-shared key. The pre-shared key can be up to 128 characters in length (e.g., including 22 characters).

When logging in the TOE will not echo passwords so that passwords are not inadvertently displayed to the user and any other users that might be able to view the login display.

Note also that should a console user have their session terminated (e.g., due to inactivity), they are required to successfully authenticate, by reentering their identity and authentication data, in order to establish a new session.

The Identification and authentication function is designed to satisfy the following security functional requirements:

- FIA_PMG_EXT.1: The TOE implements set of password composition constraints as described above.
- FIA_PSK_EXT.1: The TOE supports pre-shared keys for IPsec up to 128 characters in length.
- FIA_UAU.7: The TOE does not echo passwords as they are entered.
- FIA_UAU_EXT.2: The TOE can be configured to use external RADIUS and TACACS+ authentication servers.
- FIA_UIA_EXT.1: The TOE only displays the warning banner and allows for network switching services prior to a user being identified and authenticated.

6.5 Security management

The TOE controls user access to commands and resources based on user role. Users are given permission to access a set of commands and resources based on their user role.

The TOE includes pre-defined user roles, of which only the user roles: network-admin and level-15, are considered instances of the 'Security Administrator' or 'Authorized Administrator' as defined in the NDPP. These Security Administrator roles are capable of managing the security functions of the TOE since they allow for security relevant configuration. These capabilities include changing the user permission settings including user-role, authentication-mode, protocol, and setting the authentication password in user interface view.

The other roles represent logical subsets of those security management roles, but do not offer any security relevant configuration management capabilities. The other roles are limited to the ability to change a user's own password, non-security relevant functions and review of information. For example, the roles: network-operator, level-1 and level-9 can display the configuration and status of the TOE. The local audit log can only be accessed by those with the network-admin, network-operator, or level-15 role.

Once authenticated (none of these functions is available to any user before being identified an authenticated), authorized administrators have access to the following security functions:

- Ability to administer the TOE locally and remotely;
- Ability to update the TOE, and to verify the updates using digital signature capability prior to installing those updates;
- Ability to configure a login banner as well as network access control functions; and
- Ability to configure the cryptographic functionality

The Security management function is designed to satisfy the following security functional requirements:

- FMT_MTD.1: The TOE restricts the access to manage TSF data that can affect the security functions of the TOE to Security Administrators
- FMT_SMF.1: The TOE provides administrative interfaces to perform the functions identified above.
- FMT_SMR.2: The TOE includes 19 predefined roles. As described above only the network-admin, and level-15 roles, that have been configured to access all security management functions of the TOE corresponds to the required 'Authorized Administrator' also referred to as 'Security Administrator' in some requirements.

6.6 Protection of the TSF

The TOE is an appliance and as such is designed to work independent of other components to a large extent. Secure communication with third-party peers as addressed in section 6.8, Trusted path/channels, and secure communication among multiple instances of the TOE is limited to a direct link between clustered switch appliances. Normally

clustered components are co-located and connected via a link that would not be exposed outside of the same physical environment. As such, no additional protection (e.g., encryption) should be necessary in most operational environments.

The TOE is designed specifically to prevent access to locally-stored cryptographically protected passwords and also, while cryptographic keys can be entered, the TOE does not disclose any keys stored in the TOE. In the evaluated configuration (i.e., with FIPS mode enabled), the TOE protects user passwords either by saving a SHA-512 hash of the password (for user accounts password that existed before FIPS mode was enabled) or by encrypting the password using AES in CTR mode (for user accounts password entered after FIPS mode was enabled). Note that while some keys and passwords occur in plain text in RAM, that is only while they are in use and are not accessible by any user from RAM.

The TOE's embedded OS manages the clock and exposes administrator clock-related functions. The clock is used for audit record time stamps and measuring session activity for termination.

The TOE includes a number of built in diagnostic tests that are run during start-up to determine whether the TOE is operating properly. An administrator can configure the TOE to reboot or to stop, with errors displayed, when an error is encountered. The built-in self-tests include basic read-write memory (i.e., each memory location is written with a non-zero value and read to ensure it is stored as expected), flash read, software checksum tests, and device detection tests. When operating in CC/FIPS mode, the TOE uses CAVP tested functions to perform the power-on self-tests.

The TOE supports upgrades to the boot ROM program and system boot file as well as to support software hotfixes. The TOE provides interfaces so that an administrator can query the current boot ROM program or system boot file versions as well as to identify any installed patches. Both the boot ROM program and system boot file can be upgraded via the Boot ROM menu or the command line interface, but a reboot is required in each case. Hotfixes, which can affect only the system boot file, are installed via the command line interface and do not require a reboot to become effective.

The TOE includes a validity checking function that is enabled when upgrading the boot ROM program, while system boot files and software patches are always validated prior to installation. In each case, the upgrade version will be checked to ensure it is appropriate and the upgrade file will be verified using an embedded (HPE authorized) digital signature verified against a configured pair of hard-coded keys embedded in the TOE. If the version is incorrect or the signature cannot be verified, the upgrade will not proceed to protect the integrity of the TOE. More specifically, each update includes a header and data. The header includes a SHA-256 secure hash of the data that is signed (using rDSA/RSA 2048) by HPE. In order to verify the data, the TOE generates its own SHA-256 secure hash of the update data, compares it with the signed hash in the update header to ensure they match, and verifies the hash signature using its configured public key.

The Protection of the TSF function is designed to satisfy the following security functional requirements:

- FPT_APW_EXT.1: The TOE does not offer any functions that will disclose to any user a plain text password. Furthermore, locally defined passwords are not stored in plaintext form.
- FPT_SKP_EXT.1: The TOE does not offer any functions that will disclose to any users a stored cryptographic key
- FPT_STM.1: The TOE includes its own hardware clock.
- FPT_TST_EXT.1: The TOE includes a number of power-on diagnostics that will serve to ensure the TOE is functioning properly. The tests include ensure memory and flash can be accessed as expected, to ensure that software checksums are correct, and also to test the presence and function of plugged devices
- FPT_TUD_EXT.1: The TOE provides functions to query and upgrade the versions of the boot ROM program and system boot file (including installing hotfixes). Digital signatures are used to ensure the integrity of each upgrade prior to performing the upgrade; this checking is optional for the boot ROM program since special circumstances might require those checks to be disabled.

6.7 TOE access

The TOE is configured to display administrator-configured login banner before authentication. In all cases (console and SSH), the login banner is displayed on the display or login screen before entering the user password.

The TOE can be configured by an administrator to set a session timeout. A session (local console or remote SSH) that is inactive (i.e., no commands issuing from the remote client) for the defined timeout value will be terminated. Upon exceeding the session timeout, the TOE logs the user off.

The user will be required to login in after any session has been terminated due to inactivity or after voluntary termination. Of course, administrators can logout of local or remote sessions at any time.

The TOE access function is designed to satisfy the following security functional requirements:

- FTA_SSL.3: The TOE terminates remote sessions that have been inactive for an administrator-configured period of time.
- FTA_SSL.4: The TOE provides the function to logout (or terminate) the both local and remote user sessions as directed by the user.
- FTA_SSL_EXT.1: FTA_SSL_EXT.1: The TOE terminates local sessions that have been inactive for an administrator-configured period of time.
- FTA_TAB.1: The TOE is configured to display administrator-defined advisory banners when administrators successfully establish interactive sessions with the TOE.

6.8 Trusted path/channels

The TOE can be configured to export audit records to an external Syslog server. The TOE uses IPsec to protect communications between itself and components in the operational environment including Syslog and authentication servers (RADIUS and TACACS+).

To support secure remote administration, the TOE includes an implementation of SSHv2. An administrator with an appropriate SSHv2-capable client can establish secure remote connections with the TOE. The TOE supports both public key-based and password-based client authentication for the SSH trusted path. To successfully establish an interactive administrative session, the administrator must be able to provide acceptable user credentials (e.g., user id and password), after which they will be able to issue commands within their assigned authorizations

In all cases, the endpoints are assured by virtue of the certificates installed, trusted, and reviewable when connecting and by virtue of user authentication.

The Trusted path/channels function is designed to satisfy the following security functional requirements:

- FTP_ITC.1: In the evaluated configuration, the TOE must be configured to use IPsec to ensure that any exported audit records are sent only to the configured server and so they are not subject to inappropriate disclosure or modification. Likewise communication with authentication servers is protected via IPsec.
- FTP_TRP.1: The TOE provides SSH to support secure remote administration. Administrators can initiate a remote session that is secured (from disclosure and modification) using NIST-validated cryptographic operations, and all remote security management functions require the use of this secure channel