



Palo Alto Networks SD-WAN Instant-On Network (ION) Devices ION 1200, ION 1200-S, ION 3200, ION 5200, ION 9200, vION, and Controller running 6.5 Security Target

Version: 0.6

Date: September 9, 2026

Palo Alto Networks, Inc.

www.paloaltonetworks.com

© 2026 Palo Alto Networks, Inc. Palo Alto Networks is a registered trademark of Palo Alto Networks. A list of our trademarks can be found at <https://www.paloaltonetworks.com/company/trademarks.html>. All other marks mentioned herein may be trademarks of their respective companies.

Table of Contents

1. SECURITY TARGET INTRODUCTION	1
1.1 SECURITY TARGET, TOE AND CC IDENTIFICATION	1
1.2 CONFORMANCE CLAIMS.....	2
1.3 CONVENTIONS.....	3
1.3.1 Terminology	4
1.3.2 Acronyms.....	4
2. PRODUCT DESCRIPTION.....	6
2.1 TOE OVERVIEW	6
2.2 TOE ARCHITECTURE.....	7
2.2.1 Physical Boundaries	7
2.2.2 Logical Boundaries	10
2.3 TOE DOCUMENTATION	11
2.4 EXCLUDED FUNCTIONALITY.....	11
3. SECURITY PROBLEM DEFINITION.....	12
4. SECURITY OBJECTIVES	13
4.1 SECURITY OBJECTIVES FOR THE OPERATIONAL ENVIRONMENT	13
5. IT SECURITY REQUIREMENTS	15
5.1 EXTENDED REQUIREMENTS	15
5.2 TOE SECURITY FUNCTIONAL REQUIREMENTS	15
5.2.1 Security Audit (FAU)	16
5.2.2 Communication (FCO)	20
5.2.3 Cryptographic Support (FCS)	20
5.2.4 Identification and Authentication (FIA).....	30
5.2.5 Security Management (FMT).....	32
5.2.6 Protection of the TSF (FPT).....	34
5.2.7 TOE Access (FTA)	35
5.2.8 Trusted Path/Channels (FTP).....	35
5.3 TOE SECURITY ASSURANCE REQUIREMENTS.....	37
6. TOE SUMMARY SPECIFICATION	38
6.1 SECURITY AUDIT	38
6.2 COMMUNICATION	38
6.3 CRYPTOGRAPHIC SUPPORT	39
6.4 IDENTIFICATION AND AUTHENTICATION	49
6.5 SECURITY MANAGEMENT	51
6.6 PROTECTION OF THE TSF	53
6.7 TOE ACCESS	54
6.8 TRUSTED PATH/CHANNELS	55
7. PROTECTION PROFILE CLAIMS	56
8. RATIONALE.....	57

Figure 1: TOE Architecture 7

LIST OF TABLES

Table 1 TOE Platforms 8
Table 2 Excluded Features 11
Table 3 TOE Security Functional Components.....15
Table 4 Auditable Events 17
Table 5 Assurance Components 37
Table 6 Cryptographic Functions 39
Table 7 FIPS 186-5 Conformance 42
Table 8 Private Keys and CSPs 43

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 the Next-Generation wide area network (WAN) network devices and SD-WAN Controller running 6.5 provided by Palo Alto Networks Inc.

The SD-WAN Controller and ION Devices include the appliances shown in section 1.1. The Palo Alto Networks SD-WAN ION Devices enable the integration of a diverse set of WAN connection types, improve application performance and visibility, enhance security and compliance, and reduce the overall cost and complexity of customers' WAN. Built with the intent to reduce remote infrastructure, Palo Alto Networks SD-WAN ION devices enable the cloud-delivered branch. The ION Devices are managed by the on-premises SD-WAN Controller¹ which is included in the scope of this evaluation. The TOE which comprises SD-WAN Controller managing ION Devices is considered a Distributed TOE.

The focus of this evaluation is on the TOE functionality supporting the claims in the collaborative Protection Profile for Network Devices [NDcPP]. The only capabilities covered by the evaluation are those specified in the aforementioned Protection Profile, all other capabilities are not covered in the evaluation. The security functionality specified in [NDcPP] protection of communications between TOE components and trusted IT entities, identification and authentication of administrators, auditing of security-relevant events, ability to verify the source and integrity of updates to the TOE, the termination of IPsec VPN tunnels, and specifies ACVP-validated cryptographic mechanisms.

The Security Target contains the following additional sections:

- Product Description (Section 2)
- Security Problem Definition (Section 3)
- Security Objectives (Section 4)
- IT Security Requirements (Section 5)
- TOE Summary Specification (Section 6)
- Protection Profile Claims (Section 7)
- Rationale (Section 8).

1.1 Security Target, TOE and CC Identification

ST Title – Palo Alto Networks SD-WAN Instant-On Network (ION) Devices ION 1200, ION 1200-S, ION 3200, ION 5200, ION 9200, vION, and Controller running 6.5 Security Target

ST Version – Version 0.6

ST Date – September 9, 2026

TOE Identification – Palo Alto Networks SD-WAN appliance models include:

1. ION 1200
 - a. ION 1200
 - b. ION 1200-C-NA
 - c. ION-1200-C-ROW
 - d. ION-1200-C-5G-WW
2. ION 1200-S
 - a. ION 1200-S

¹ Cloud-based SD-WAN Controller is vendor affirmed and outside the scope of evaluation.

- b. ION 1200-S-C-NA
 - c. ION 1200-S-C-ROW
 - d. ION 1200-S-C-5G-WW
- 3. ION 3200
 - a. ION 3200
 - b. ION 3200H
- 4. ION 5200
 - a. ION 5200
- 5. ION 9200
 - a. ION 9200
- 6. Virtual ION (vION)
- 7. SD-WAN Controller running on Ubuntu 24.04

Evaluation testing for Palo Alto Networks SD-WAN Instant-On Network (ION) Devices ION 1200, ION 1200-S, ION 3200, ION 5200, ION 9200, vION, and Controller running 6.5 included the following:

Controller included the following hardware and processors:

VMware ESXi 8.0 Update 3:

- Dell PowerEdge (PE) R740 Processor: Intel Xeon Gold 6248 (Cascade Lake microarchitecture) with Broadcom 57416 NIC
- Memory: 128 GB RDIMM

Devices included the following hardware and processors:

- ION 9200: Intel Atom P5362 processor (Snow Ridge)
- vION
 - VMware ESXi 8.0 Update 3:
 - Dell PowerEdge (PE) R740 Processor: Intel Xeon Gold 6248 (Cascade Lake microarchitecture) with Broadcom 57416 NIC
 - Memory: 128 GB RDIMM

TOE Developer – Palo Alto Networks, Inc.

Evaluation Sponsor – Palo Alto Networks, Inc.

CC Identification – *Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5, April 2017*

1.2 Conformance Claims

This ST and the TOE it describes are conformant to the following CC specifications:

- collaborative Protection Profile for Network Devices, Version 3.0e, 6 December 2023 [NDcPP]
- Functional Package for Secure Shell (SSH), Version 1.0, 13 May 2021 [SSH10]

The following NIAP Technical Decisions apply to [NDcPP/SSH10] and have been accounted for in the ST development and the conduct of the evaluation:

Package	Technical Decision	Applied	Notes
CPP_ND_V3.0E	TD0836 - NIT Technical Decision: Redundant Requirements in FPT_TST_EXT.1	Yes	
CPP_ND_V3.0E	TD0868 - NIT Technical Decision: Clarification of time frames in FCS_IPSEC_EXT.1.7 and FCS_IPSEC_EXT.1.8	Yes	
CPP_ND_V3.0E	TD0879 - NIT Decision: Correction of Chapter Headings in CPP_ND_V3.0E	Yes	
CPP_ND_V3.0E	TD0880 - NIT Decision: Removal of Duplicate Selection in FMT_SMF.1.1	Yes	
CPP_ND_V3.0E	TD0886 - Clarification to FAU_STG_EXT.1 Test 6	Yes	
CPP_ND_V3.0E	TD0899 - NIT Technical Decision: Correction of Renegotiation Test for TLS 1.2	Yes	
CPP_ND_V3.0E	TD0900 - NIT Technical Decision: Clarification to Local Administrator Access in FIA_UIA_EXT.1.3	Yes	
CPP_ND_V3.0E	TD0921 - NIT Technical Decision: Addition of FIPS PUB 186-5 and Correction of Assignment	Yes	
CPP_ND_V3.0E	TD0923 - NIT Technical Decision: Auditable event for FAU_STG_EXT.1 in FAU_GEN.1.2	Yes	
CPP_ND_V3.0E	TD0973 - NIT Technical Decision: FCS_(D)TLSS_EXT.1.3 Test 2 DHE Ciphersuite Conditionality	Yes	
CPP_ND_V3.0E	TD0990 - NIT Technical Decision: CTR_DRBG in FCS_RBG_EXT.1.2	Yes	
CPP_ND_V3.0E	TD1033 - Sunset Dates for NDcPP Configurations	Yes	
CPP_ND_V3.0E	TD1052 - NIT Technical Decision: Separation of Test Definitions for DTLSv1.2 and v1.3 (Renegotiation)	Yes	
PKG_SSH_V1.0	TD0682- Addressing Ambiguity in FCS_SSHS_EXT.1 Tests	Yes	
PKG_SSH_V1.0	TD0695 - Choice of 128 or 256 bit size in AES-CTR in SSH Functional Package	Yes	
PKG_SSH_V1.0	TD0732 - FCS_SSHS_EXT.1.3 Test 2 Update	Yes	
PKG_SSH_V1.0	TD0777 - Clarification to Selections for Auditable Events for FCS_SSH_EXT.1	Yes	
PKG_SSH_V1.0	TD0909 - Updates to FCS_SSH_EXT.1.1 App Note in SSH FP 1.0	Yes	
PKG_SSH_V1.0	TD0967 - Allowance of Kex-strict in PKG_SSH_V1.0	Yes	
PKG_SSH_V1.0	TD1015 - Addition of ECD to PKG_SSH_V1.0	Yes	
PKG_SSH_V1.0	TD1023 - Clarifications to FCS_SSH_EXT.1.5 When aes256-gcm@openssh.com Is Selected	Yes	

Common Criteria for Information Technology Security Evaluation Part 2: Security functional components, Version 3.1, Revision 5, April 2017.

- Part 2 Extended

Common Criteria for Information Technology Security Evaluation Part 3: Security assurance components, Version 3.1 Revision 5, April 2017.

- Part 3 Conformant.

1.3 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.

- All operations performed in this ST are identified according to conventions described in [NDcPP/SSH10].
- The ST author does not change operations that have been completed by the PP authors nor undo the formatting. For example, if the text is italicized, bolded, or underlined by the PP author, the ST author will not undo it. In this way operations have been identified.
- Selection/Assignment operations completed by the PP author remain as described in the [NDcPP/SSH10].
- Selection/Assignment operations completed by the ST author were bolded to show that it was completed by the ST author and not taken as-is from the PP.
- Iteration operations completed by the ST author are identified with (1), (2), and (next number) with descriptive text following the name (e.g. FCS_HTTPS_EXT.1(1) HTTPS Protocol (TLS Server)).
- Refinement operations completed by the ST author are identified in BOLD text.

1.3.1 Terminology

The following terms and abbreviations are used in this ST:

Controller	SD-WAN appliances obey operational regulations passed down from the central SD-WAN controller. This significantly reduces or eliminates the need to manage gateways and routers individually.
SD-WAN	SD-WAN (software-defined wide area network) is a type of networking technology that uses software-defined networking (SDN) principles to manage and optimize the performance of wide area networks (WANs).

1.3.2 Acronyms

AES	Advanced Encryption Standard
CBC	Cipher-Block Chaining
CC	Common Criteria
CEM	Common Evaluation Methodology
CM	Configuration Management
DH	Diffie-Hellman
DRBG	Deterministic Random Bit Generator
DP	Data Plane
EEPROM	Electrically Erasable Programmable Read-Only Memory
FIPS	Federal Information Processing Standard
FSP	Functional Specification
FW	Firewall
GCM	Galois/Counter Mode
GUI	Graphical User Interface
HMAC	Hashed Message Authentication Code
HTTPS	Hypertext Transfer Protocol Secure
ICMP	Internet Control Message Protocol
IKE	Internet Key Exchange
IoT	Internet of Things
IP	Internet Protocol
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
IPsec	Internet Protocol Security
MP	Management Plane
MPLS	Multiprotocol Label Switching
NAT	Network Address Translation
NIST	National Institute of Standards and Technology
PP	Protection Profile

RSA	Rivest, Shamir and Adleman (algorithm for public-key cryptography)
SA	Security Association
SaaS	Software as a Service
SAR	Security Assurance Requirement
SFR	Security Functional Requirement
SHA	Secure Hash Algorithm
SLA	Service Level Agreement
SSL	Secure Socket Layer
ST	Security Target
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Functions
UDP	User Datagram Protocol
URL	Uniform Resource Locator
VLAN	Virtual Local Area Network
VM	Virtual Machine
VPN	Virtual Private Network
VPNGW	Virtual Private Network Gateway
WAN	Wide Area Network

2. Product Description

The Palo Alto next-generation SD-WAN appliances are network appliances on specialized physical hardware used to manage enterprise network traffic flow using function-specific processing for networking, security, and management. The next-generation ION Devices are managed by SD-WAN Controller to deliver centralized management and flexible connectivity. The following benefits are provided for completeness only and are not meant to be statement of endorsement or marketing:

- **Elastic networks:** Delivering a zero-routing network with a centralized controller-based architecture that seamlessly builds, manages, and updates the network while flexibly supporting carrier-independent WAN connections like 5G, MPLS, broadband, and satellite.
- **App-defined fabric:** Providing direct-to-app access that ensures an exceptional user experience for all applications like SaaS, cloud, and business-critical/private applications while delivering application availability based on application performance SLAs.
- **Zero Trust security:** Providing integrated and cloud-delivered security services to branch offices. The security needs to be granular (Layer 7) to enforce true least-privileged access and ensure only the right people get access to the right information and assets, including IoT.
- **AI-powered operations:** Leveraging the latest advancements in observability and AI/ML built natively to help customers automate complex IT and network operations center (NOC) functions with the power of AI/ML to increase productivity and reduce mean time to repair (MTTR).

2.1 TOE Overview

The Target of Evaluation (TOE) is comprised of one or more Palo Alto Networks next-generation ION Devices that include the Palo Alto Networks ION 1200, ION 1200-S, ION 3200, ION 5200, ION 9200, or vION managed by a SD-WAN Controller all running 6.5. The next-generation ION devices are deployed in the network at the WAN edge and automatically begin examining application data on the network to identify the application and measure several key performance indicators of each session. Statistics from your network are stored securely in the SD-WAN controller, which can be used to configure ION devices, define applications and sites, and monitor end-to-end application performance and availability.

Centralized Management

The SD-WAN Controller provides secure connections for the Web Management interface. The Web interfaces provide authorized administrators with the ability to manage, configure and monitor the ION Devices.

Compliant Mode of Operation

The TOE is compliant with the capabilities outlined in this Security Target only when operated in FIPS-Common Criteria mode (also referred to as FIPS-CC mode). FIPS-CC mode is a special operational mode in which the FIPS and CC requirements for self-tests as well as X.509v3 certificates checks are enforced. In this mode, only CC Approved cryptographic algorithms and key sizes are available.

2.2 TOE Architecture

The TOE's architecture is divided into two main TOE components: the SD-WAN Controller and the ION Device(s). The Controller provides system management functionality while the ION Devices handle all data processing on the network. The ION Devices can optionally connect to third party VPN devices using IKE/IPsec protocol. NOTE: The ION Device does have a CLI SSH interface used for troubleshooting only and not intended for daily management. In the evaluated configuration, the ION Devices are managed by the SD-WAN Controller.

The following diagram depicts both the SD-WAN Controller and ION Devices:

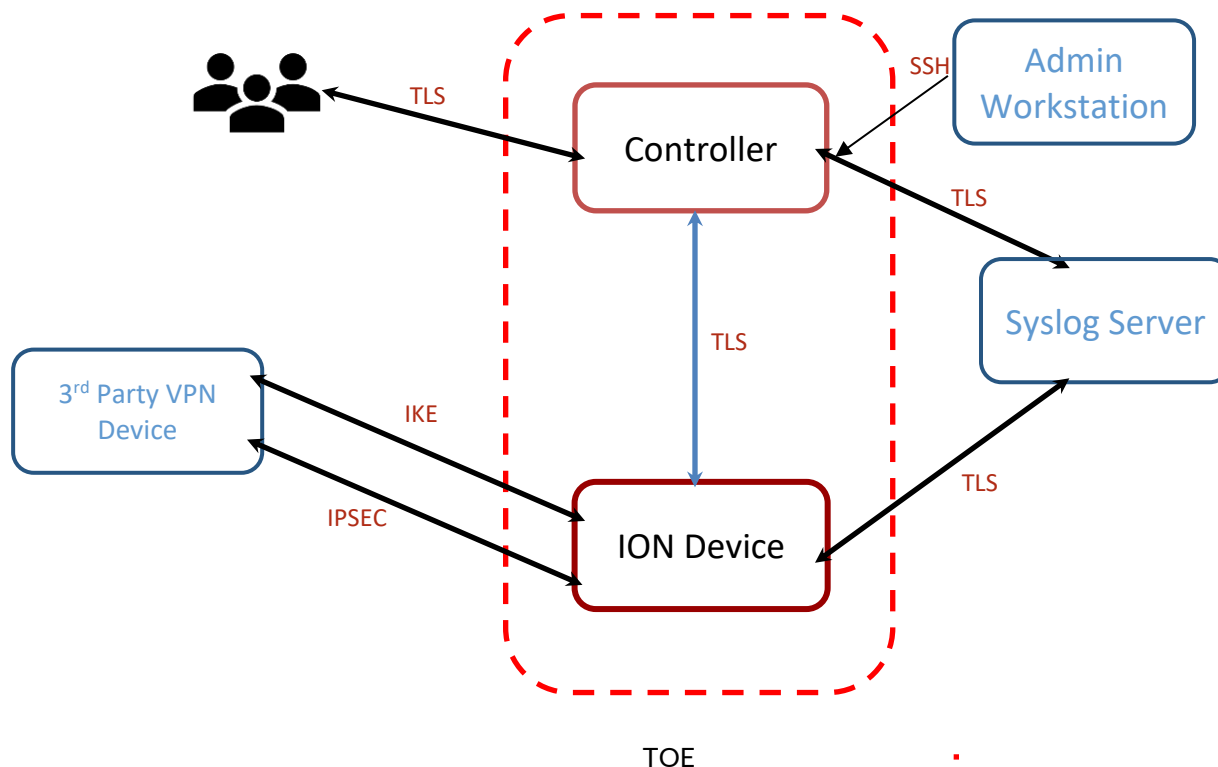


Figure 1: TOE Architecture

2.2.1 Physical Boundaries

The TOE consists of the following components:

- Hardware appliance-includes the physical port connections on the outside of the appliance cabinet and a time clock from NTP server that provides the time stamp used for the audit records.
- On-Premises Controller is a virtualized software component that can be installed on any operating system or hypervisor with Java Virtual Machine.

The physical boundary of the TOE comprises the ION appliance (ION 1200, ION 1200-S, ION 3200, ION 5200, ION 9200, vION) and SD-WAN Controller. The next-generation appliance models differ in their performance capability, but they provide the same security functionality.

The ION appliance attaches to a physical network and includes the following ports and processors:

- ION 1200: 4 RJ-45 10/100/1000 ports for network traffic (Ethernet ports); 1 RJ-45 port for connecting a serial console (management console port); 2 USB (disabled in FIPS-CC mode except for power) and Power input. Processor: Intel Atom C3436L (Denverton)
- ION 1200-S: 6 RJ-45 10/100/1000 ports for network traffic (Ethernet ports); 2 (Giga Ethernet) GE RJ45/SFP Combo ports, 2 GE RJ45 bypass ports, 4 (Power over Ethernet) POE++ ports; 1 RJ-45 port for connecting a serial console (management console port); 2 USB (disabled in FIPS-CC mode except for power) and Power input. Processor: Intel Atom C3436L (Denverton)
- ION 3200/3200H: 6 RJ-45 10/100/1000 ports for network traffic (Ethernet ports); 2 (Giga Ethernet) GE RJ45/SFP Combo ports, 2 GE RJ45 bypass ports, 4 (Power over Ethernet) POE++ ports; 1 RJ-45 port for connecting a serial console (management console port); 2 USB (disabled in FIPS-CC mode except for power) and Power input. Processor: Intel Atom C3558R (Denverton)
- ION 5200: 10 RJ-45 10/100/1000 ports for network traffic (Ethernet ports); 4 (Giga Ethernet) GE RJ45/SFP Combo ports, 4 GE RJ45 bypass ports, 4 (Power over Ethernet) POE++ ports; 1 RJ-45 port for connecting a serial console (management console port); 2 USB (disabled in FIPS-CC mode except for power) and Power input. Processor: Intel Atom C5325 (Parker Ridge)
- ION 9200: 10 RJ-45 10/100/1000 ports for network traffic (Ethernet ports); 4 (Giga Ethernet) GE RJ45/SFP Combo ports, 8 GE RJ45 bypass ports, 4 (Power over Ethernet) POE++ ports; 1 RJ-45 port for connecting a serial console (management console port); 2 USB (disabled in FIPS-CC mode except for power) and Power input. Processor: Intel Atom P5362 (Snow Ridge)

In the evaluated configuration, the SD-WAN Controller can be accessed by:

- A computer securely connected to the network port via an RJ-45 Ethernet cable. The management channel that provides access to the Web GUI is protected via HTTPS (HTTP over TLS). The computer is part of the operational environment and required to have a web browser (for accessing the GUI).

The operational environment includes the following:

- Syslog server
- NTP server
- VPN gateway peer(s)
- Workstation
 - Web browsers

Table 1 TOE Platforms

Product Identification	Illustration	Description
ION 1200		• Throughput (encrypted 1,400 byte packets): 700 Mbps • Throughput (encrypted 600 byte packets): 250 Mbps • Cellular Support: 4G LTE/5G • DC VPN Scale: N/A • Use Case: Enterprise small branch

Product Identification	Illustration	Description
ION 1200-S		- Throughput (encrypted 1,400 byte packets): 700 Mbps - Throughput (encrypted 600 byte packets): 250 Mbps - Cellular Support: 4G LTE/5G - DC VPN Scale: N/A - Use Case: Enterprise small branch
ION 3200		- Throughput (encrypted 1,400 byte packets): 1.5 Gbps (DC), 1 Gbps (Branch) - Throughput (encrypted 600 byte packets): 600 Mbps (DC), 500 Mbps (Branch) - Cellular Support: N/A - DC VPN Scale: 500 - Use Case: Enterprise small branch, data center (DC)
ION 3200H		- Throughput (encrypted 1,400 byte packets): 1.5 Gbps (DC), 1 Gbps (Branch) - Throughput (encrypted 600 byte packets): 600 Mbps (DC), 500 Mbps (Branch) - Cellular Support: N/A - DC VPN Scale: 500 - Use Case: Enterprise small branch, data center (DC)
ION 5200		- Throughput (encrypted 1,400 byte packets): 4 Gbps (DC), 2 Gbps (Branch) - Throughput (encrypted 600 byte packets): 1.5 Gbps (DC), 1 Gbps (Branch) - Cellular Support: N/A - DC VPN Scale: 2,000 - Use Case: Enterprise large branch, data center
ION 9200		- Throughput (encrypted 1,400 byte packets): 15 Gbps (DC), 8 Gbps (Branch) - Throughput (encrypted 600 byte packets): 6 Gbps (DC), 3 Gbps (Branch) - Cellular Support: N/A - DC VPN Scale: 5,000 - Use Case: Multigigabit remote office data center and large campus

2.2.2 Logical Boundaries

This section summarizes the security functions provided by the TOE:

- Security Audit
- Communication
- Cryptographic Support
- Identification and Authentication
- Security Management
- Protection of the TSF
- TOE Access
- Trusted Path/Channels

2.2.2.1 Security Audit

The TOE is designed to be able to generate logs with the required content (e.g., date/time, username, event type, etc.) for a wide range of security relevant events including the events specified in [NDcPP/SSH10]. The TOE can be configured to store the logs locally and be configured to send the logs securely to a designated external log server.

2.2.2.2 Communication

The TOE is a Distributed TOE with secure registration to communicate with and join TOE components together. Security Administrator can remove any ION Device from the SD-WAN Controller as well.

2.2.2.3 Cryptographic Support

The TOE implements NIST-validated cryptographic algorithms that provide key management, random bit generation, encryption/decryption, digital signature and cryptographic hashing and keyed-hash message authentication features in support of higher-level cryptographic protocols, including IKE/IPsec, SSH, HTTPS, and TLS. Note that to be in the evaluated configuration, the TOE must be configured in FIPS-CC mode, which ensures the TOE's configuration is compliant with the FIPS and NIST standards.

2.2.2.4 Identification and Authentication

The TOE requires all users accessing the TOE user interfaces to be successfully identified and authenticated before they can access any security management functions available in the TOE. The TOE (i.e., Controller) offers network accessible (HTTPS) connections to the GUI.

The TOE supports the local (i.e., on device) definition and authentication of administrators with username, password, and role (set of privileges), which it uses to authenticate the user and to associate that user with an authorized role.

2.2.2.5 Security Management

The TOE provides a GUI to access the security management functions. Security management commands are limited to administrators and are available only after they have provided acceptable user identification and authentication data to the TOE. The TOE provides access to the GUI using an HTTPS/TLS supported web browser.

The TOE provides a number of management functions and restricts them to users with the appropriate privileges. The management functions include the capability to configure the login banner, configure the idle timeout, configure IKE/IPsec VPN, configure NTP server, and other management functions. The administrator role is considered as Security Administrator as defined in the [NDcPP/SSH10] for the purposes of this ST.

2.2.2.6 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. It also provides a timing mechanism (NTP) to ensure that reliable time information is available (e.g., for log accountability).

The TOE includes functions to perform self-tests so that it can detect when it is failing and transition to a secure, maintenance state. It also includes a mechanism to verify TOE updates to prevent malicious or other unexpected changes in the TOE.

2.2.2.7 TOE Access

The TOE can be configured to display an administrator-defined advisory banner before establishing an administrative user session and to terminate remote interactive sessions after a configurable period of inactivity. It also provides users the capability to terminate their own interactive sessions.

2.2.2.8 Trusted Path/Channels

The TOE protects interactive communication with remote administrators using HTTP over TLS (HTTPS) or SSH (to the Controller). TLS, and IKE/IPsec ensure both integrity and disclosure protection.

The TOE protects communication with external log server with TLS; and remote VPN gateways/peers using IKE/IPsec to prevent unintended disclosure or modification of the transferred data.

2.3 TOE Documentation

Palo Alto Networks Inc. offers a series of documents that describe the installation of Palo Alto Networks SD-WAN Devices as well as guidance for subsequent use and administration of the applicable security features.

For Palo Alto Networks SD-WAN Instant-On Network (ION) Devices ION 1200, ION 1200-S, ION 3200, ION 5200, ION 9200, vION, and Controller running 6.5., these documents include:

- Prisma SD-WAN Administrator's Guide
- Palo Alto Networks Common Criteria Evaluated Configuration Guide (CCECG)

2.4 Excluded Functionality

The list below identifies features or protocols that are not evaluated or must be disabled, and the rationale why. Note that this does not mean the features cannot be used in the evaluated configuration (unless explicitly stated so). It means that the features were not evaluated and/or validated by an independent third party and the functional correctness of the implementation is vendor assertion. Evaluated functionality is scoped exclusively to the security functional requirements specified in Security Target. In particular, only the following protocols implemented by the TOE have been tested, and only to the extent specified by the security functional requirements: TLS, HTTPS, IKE/IPsec.

Table 2 Excluded Features

Feature	Description
Any features not associated with SFRs in claimed [NDcPP/SSH10]	NDcPP/SSH10 forbids adding additional requirements to the Security Target (ST). If additional functionalities or products are mentioned in the ST, it is for completeness only.

3. Security Problem Definition

This security target includes by reference the Security Problem Definition (composed of organizational policies, threat statements, and assumption) from [NDcPP/SSH10].

In general, the [NDcPP/SSH10] has presented a Security Problem Definition appropriate for network infrastructure devices, such as firewalls, routers, managers and as such is applicable to the Palo Alto TOE.

4. Security Objectives

Like the Security Problem Definition, this security target includes by reference the Security Objectives from the [NDcPP/SSH10]. The security objectives for the operational environment are reproduced below since these objectives characterize technical and procedural measures each consumer must implement in their operational environment.

In general, the [NDcPP/SSH10] has presented Security Objectives appropriate for network infrastructure devices, such as is applicable to the Palo Alto TOE.

4.1 Security Objectives for the Operational Environment

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

platform on which the VM runs is removed from its operational environment.

OE.COMPONENTS_RUNNING

For distributed TOEs, the Security Administrator ensures that the availability of every TOE component is checked as appropriate to reduce the risk of an undetected attack on (or failure of) one or more TOE components. The Security Administrator also ensures that it is checked as appropriate for every TOE component that the audit functionality is running properly.

OE.VM_CONFIGURATION

For vNDs, the Security Administrator ensures that the VS and VMs are configured to

- reduce the attack surface of VMs as much as possible while supporting ND functionality (e.g., remove unnecessary virtual hardware, turn off unused inter-VM communications mechanisms), and
- correctly implement ND functionality (e.g., ensure virtual networking is properly configured to support network traffic, management channels, and audit reporting).

The VS should be operated in a manner that reduces the likelihood that vND operations are adversely affected by unauthorized features such as cloning, save/restore, suspend/resume, and live migration.

If possible, the VS should be configured to make use of features that leverage the VS's privileged position to provide additional security functionality. Such features could include malware detection through VM introspection, measured VM boot, or VM snapshot for forensic analysis.

5. IT 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 [NDcPP/SSH10].

The SARs are the set of SARs specified for Network Devices.

5.1 Extended Requirements

All the extended requirements in this ST have been drawn from the [NDcPP/SSH10]. The [NDcPP/SSH10] defines all the extended SFRs (*_EXT) and since they are not redefined in this ST, this PP/package should be consulted for more information regarding those CC extensions.

5.2 TOE Security Functional Requirements

The following table identifies the SFRs that are satisfied by the Palo Alto TOE.

Table 3 TOE Security Functional Components

Requirement Class	Requirement Component	Location
FAU: Security Audit	FAU_GEN.1: Audit Data Generation	NDcPP
	FAU_GEN.2: User Identity Association	NDcPP
	FAU_STG_EXT.1: Protected Audit Event Storage	NDcPP
	FAU_STG.1: Protected Audit Trail Storage	NDcPP
FCO: Communication	FCO_CPC_EXT.1: Component Registration Channel Definition	NDcPP
FCS: Cryptographic Support	FCS_CKM.1: Cryptographic Key Generation	NDcPP
	FCS_CKM.2: Cryptographic Key Establishment	NDcPP
	FCS_CKM.4: Cryptographic Key Destruction	NDcPP
	FCS_COP.1/DataEncryption: Cryptographic Operation (AES Data Encryption/Decryption)	NDcPP
	FCS_COP.1/SigGen: Cryptographic Operation (Signature Generation and Verification)	NDcPP
	FCS_COP.1/Hash: Cryptographic Operation (Hash Algorithm)	NDcPP
	FCS_COP.1/KeyedHash: Cryptographic Operation (Keyed Hash Algorithm)	NDcPP
	FCS_RBG_EXT.1: Random Bit Generation	NDcPP
	FCS_HTTPS_EXT.1: HTTPS Protocol	NDcPP
	FCS_TLSC_EXT.1: TLS Client Protocol	NDcPP
	FCS_TLSS_EXT.1: TLS Server Protocol	NDcPP
	FCS_IPSEC_EXT.1: IPsec Protocol	NDcPP
	FCS_NTP_EXT.1: NTP Protocol	NDcPP
	FCS_SSH_EXT.1: SSH Protocol	SSH10

Requirement Class	Requirement Component	Location
	FCS_SSHS_EXT.1: SSH Protocol - Server	SSH10
FIA: Identification and Authentication	FIA_AFL.1: Authentication Failure Management	NDcPP
	FIA_PMG_EXT.1: Password Management	NDcPP
	FIA_UIA_EXT.1: User Identification and Authentication	NDcPP
	FIA_UAU.7: Protected Authentication Feedback	NDcPP
	FIA_X509_EXT.1/ITT: X.509 Certificate Validation	NDcPP
	FIA_X509_EXT.1/Rev: X.509 Certificate Validation	NDcPP
	FIA_X509_EXT.2: X.509 Certificate Authentication	NDcPP
	FIA_X509_EXT.3: X.509 Certificate Requests	NDcPP
FMT: Security Management	FMT_MOF.1/ManualUpdate: Management of Security Functions Behaviour	NDcPP
	FMT_MOF.1/Services: Management of Security Functions Behaviour	NDcPP
	FMT_MTD.1/CryptoKeys: Management of TSF Data	NDcPP
	FMT_MTD.1/CoreData: Management of TSF Data	NDcPP
	FMT_SMF.1: Specification of Management Functions	NDcPP
	FMT_SMR.2: Restrictions on Security Roles	NDcPP
FPT: Protection of the TSF	FPT_SKP_EXT.1: Protection of TSF Data (for reading of all symmetric keys)	NDcPP
	FPT_APW_EXT.1: Protection of Administrator Passwords	NDcPP
	FPT_STM_EXT.1: Reliable Time Stamps	NDcPP
	FPT_TST_EXT.1: TSF Testing	NDcPP
	FPT_TUD_EXT.1: Trusted Update	NDcPP
	FPT_ITT.1: Basic Internal TSF Data Transfer Protection	NDcPP
FTA: TOE Access	FTA_SSL.3: TSF-initiated Termination	NDcPP
	FTA_SSL.4: User-initiated Termination	NDcPP
	FTA_SSL_EXT.1: TSF-initiated Session Locking	NDcPP
	FTA_TAB.1: Default TOE Access Banners	NDcPP
FTP: Trusted Path/Channels	FTP_ITC.1: Inter-TSF Trusted channel	NDcPP
	FTP_TRP.1/Admin: Trusted Path	NDcPP

5.2.1 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:

- Start-up and shutdown of the audit functions;
- All auditable events for the not specified level of audit; and
- All administrative actions comprising:*

- Administrative login and logout (name of Administrator account shall be logged if individual accounts are required for Administrators).
 - Changes to TSF data related to configuration changes (in addition to the information that a change occurred it shall be logged what has been changed).
 - Generating/import of, changing, or deleting of cryptographic keys (in addition to the action itself a unique key name or key reference shall be logged).
 - **[Resetting passwords (name of related user account shall be logged)];**
- d) Specifically defined auditable events listed in Table 4

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

- a) Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and
- b) For each audit event type, based on the auditable event definitions of the functional components included in the cPP/ST, information specified in column three of Table 4.

Table 4 Auditable Events

Requirement	Auditable Events	Additional Audit Record Contents
FAU_GEN.1	None.	None.
FAU_GEN.2	None.	None.
FAU_STG_EXT.1	Configuration of local audit settings.	Identity of account making changes to the audit configuration.
FAU_STG.1	None.	None.
FCO_CPC_EXT.1	• Enabling communications between a pair of components. • Disabling communications between a pair of components.	Identities of the endpoint pairs enabled or disabled.
FCS_CKM.1	None.	None.
FCS_CKM.2	None.	None.
FCS_CKM.4	None.	None.
FCS_COP.1/DataEncryption	None.	None.
FCS_COP.1/SigGen	None.	None.
FCS_COP.1/Hash	None.	None.
FCS_COP.1/KeyedHash	None.	None.
FCS_RBG_EXT.1	None.	None.
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
FCS_NTP_EXT.1	• Configuration of a new time server • Removal of configured time server	Identity if new/removed time server
FCS_SSH_EXT.1	• Failure to establish SSH connection • Establishment of SSH connection	• Reason for failure. Non-TOE endpoint of attempted connection (IP Address).

Requirement	Auditable Events	Additional Audit Record Contents
	- Termination of SSH connection session - None	- Non-TOE endpoint of attempted connection (IP Address). - Non-TOE endpoint of attempted connection (IP Address). - None
FCS_SSHS_EXT.1	None	
FCS_TLSC_EXT.1	Failure to establish a TLS session.	Reason for failure
FCS_TLSS_EXT.1	Failure to establish a TLS session.	Reason for failure
FCS_TLSC_EXT.2	None	None
FCS_TLSS_EXT.2	Failure to authenticate the client	Reason for failure
FIA_AFL.1	Unsuccessful login attempts limit is met or exceeded.	Origin of the attempt (e.g., IP address)
FIA_PMG_EXT.1	None.	None.
FIA_UIA_EXT.1	All use of identification and authentication mechanisms.	Origin of the attempt (e.g., IP address).
FIA_UAU.7	None.	None.
FIA_X509_EXT.1/ITT	- Unsuccessful attempt to validate a certificate - Any addition, replacement or removal of trust anchors in the TOE's trust store	- Reason for failure of certificate validation - Identification of certificates added, replaced or removed as trust anchor in the TOE's trust store
FIA_X509_EXT.1/Rev	- Unsuccessful attempt to validate a certificate - Any addition, replacement or removal of trust anchors² in the TOE's trust store	- Reason for failure of certificate validation - Identification of certificates added, replaced or removed as trust anchor in the TOE's trust store
FIA_X509_EXT.2	None.	None.
FIA_X509_EXT.3	None.	None.
FMT_MOF.1/ManualUpdate	Any attempt to initiate a manual update	None.
FMT_MOF.1/Services	None.	None.
FMT_MTD.1/CryptoKeys	None.	None.
FMT_MTD.1/CoreData	None.	None.
FMT_SMF.1	All management activities of TSF data	None.
FMT_SMR.2	None.	None.

² Importing CA certificate(s) or generating CA certificate(s) internally will implicitly set them as trust anchor.

Requirement	Auditable Events	Additional Audit Record Contents
FPT_SKP_EXT.1	None.	None.
FPT_APW_EXT.1	None.	None.
FPT_TST_EXT.1	None.	None.
FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure)	None.
FPT_ITT.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.
FPT_STM_EXT.1	Discontinuous changes to time – either Administrator actuated or changed via an automated process. (Note that no continuous changes to time need to be logged. See also application note on FPT_STM_EXT.1)	For discontinuous changes to time: The old and new values for the time. Origin of the attempt to change time for success and failure (e.g., IP address).
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.	- None - None - Reason for failure
FTP_TRP.1/Admin	- Initiation of the trusted path. - Termination of the trusted path. - Failures of the trusted path functions.	- None - None - Reason for failure

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 – Protected Audit Event Storage

- FAU_STG_EXT.1.1** The TSF shall be able to transmit the generated audit data to an external IT entity using a trusted channel according to FTP_ITC.1.
- FAU_STG_EXT.1.2** The TSF shall be able to store generated audit data on the TOE itself. In addition [
- ***The TOE shall be a distributed TOE that stores audit data on the following TOE components: [SD-WAN Controller, ION Devices].***
- FAU_STG_EXT.1.3** The TSF shall maintain a [log file] of audit records in the event that an interruption of communication with the remote audit server occurs.
- FAU_STG_EXT.1.4** The TSF shall be able to store [persistent] audit records locally with a minimum storage size of [45 MB].
- FAU_STG_EXT.1.5** The TSF shall [overwrite previous audit records according to the following rule: [overwrite oldest records first]] when the local storage space for audit data is full.
- FAU_STG_EXT.1.6** The TSF shall provide the following mechanisms for administrative access to locally stored audit records [none].

FAU_STG.1 – Protected Audit Trail Storage

- FAU_STG.1.1** The TSF shall protect the stored audit records in the audit trail from unauthorized deletion.
- FAU_STG.2.1** The TSF shall be able to prevent unauthorized modifications to the stored audit records in the audit trail.

5.2.2 Communication (FCO)**FCO_CPC_EXT.1 – Component Registration Channel Definition**

- FCO_CPC_EXT.1.1** The TSF shall require a Security Administrator to enable communications between any pair of TOE components before such communication can take place.
- FCO_CPC_EXT.1.2** The TSF shall implement a registration process in which components establish and use a communications channel that uses [
- ***A channel that meets the secure channel requirements in [FPT_ITT.1]]***
- for at least TSF data.
- FCO_CPC_EXT.1.3** The TSF shall enable a Security Administrator to disable communications between any pair of TOE components.

5.2.3 Cryptographic Support (FCS)**FCS_CKM.1 – Cryptographic Key Generation**

- FCS_CKM.1.1** The TSF shall generate **asymmetric** cryptographic keys in accordance with a specified cryptographic key generation algorithm: [
- ***RSA schemes using cryptographic key sizes of [2048, 3072, 4096 bits] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3 or FIPS PUB 186-5, "Digital Signature Standard (DSS)", A.1;***

- *ECC schemes using “NIST curves” [P-256, P-384, P-521] that meet the following: FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.4, or FIPS PUB 186-5, “Digital Signature Standard (DSS)”, Appendix A.2, or ISO/IEC 14888-3, “IT Security techniques - Digital signatures with appendix - Part 3: Discrete logarithm based mechanisms”, Section 6.6;*
- *FFC Schemes using ‘safe-prime’ groups that meet the following: “NIST Special Publication 800-56A Revision 3, Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography” and [RFC 3526, RFC7919].*

~~] and specified cryptographic key sizes [assignment: cryptographic key sizes] that meet the following: [assignment: list of standards].112 bits.~~

FCS_CKM.2 – Cryptographic Key Establishment

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

- *RSA-based key establishment schemes that meet the following: RSAES-PKCS1-v1_5 as specified in Section 7.2 of RFC 8017, “Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.2,*
- *Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 3, “Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography”;*
- *FFC Schemes using “safe-prime” groups that meet the following: NIST Special Publication 800-56A Revision 3, “Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography” and [groups listed in RFC 3526, groups listed in RFC 7919].*

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

FCS_CKM.4 – Cryptographic Key Destruction

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method

- *For plaintext keys in volatile storage, the destruction shall be executed by a [single overwrite consisting of [a pseudo-random pattern using the TSF’s RBG]];*
- *For plaintext keys in non-volatile storage, the destruction shall be executed by the invocation of an interface provided by a part of the TSF that [*
 - *logically addresses the storage location of the key and performs a [[three]-pass] overwrite consisting of [[different alternating patterns that does not contain any CSP]];*

that meets the following: No Standard.

FCS_COP.1/DataEncryption – Cryptographic Operation (AES Data Encryption/Decryption)

FCS_COP.1.1/DataEncryption The TSF shall perform encryption/decryption in accordance with a specified cryptographic algorithm AES used in [GCM, CBC] and [CTR] mode and cryptographic key sizes [128 bits, 256 bits], and [192 bits] that meet the following: AES as specified in ISO 18033-3, [CBC as specified in ISO 10116, GCM as specified in ISO 19772] and [CTR as specified in ISO 10116].

Application Note 2: The 192 bits AES key size is used in CBC mode only and for IPsec only.

FCS_COP.1/SigGen – Cryptographic Operation (Signature Generation and Verification)

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

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

] and cryptographic key sizes [

- **For RSA: [2048, 3072, or 4096 bits],**
- **For ECDSA: [256, 384, or 521 bits]**

] that meet the following: [

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

].

TD0921 applied

FCS_COP.1/Hash – Cryptographic Operation (Hash Algorithm)

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

FCS_COP.1/KeyedHash – Cryptographic Operation (Keyed Hash Algorithm)

FCS_COP.1.1/KeyedHash The TSF shall perform *keyed-hash message authentication* in accordance with a specified cryptographic algorithm [**HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512, implicit**] and cryptographic key sizes [**256, 384, 512**] and **message digest sizes [256, 384, 512] bits** that meet the following: ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2".

FCS_RBG_EXT.1 – Random Bit Generation

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

FCS_RBG_EXT.1.2 The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [**[one] platform-based noise source**] with a minimum of [**256 bits**] of entropy at least equal to the greatest security strength, according to ISO/IEC 18031:2011 Table C.1 "Security Strength Table for Hash Functions", of the keys and hashes that it will generate.

FCS_HTTPS_EXT.1 – HTTPS Protocol

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

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

FCS_IPSEC_EXT.1 – IPsec Protocol

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 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 [tunnel 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 (specified in RFC 3602), AES-GCM-128, AES-GCM-256 (specified in RFC 4106)] and [AES-CBC-192 (specified in RFC 3602)] together with a Secure Hash Algorithm (SHA)-based HMAC [HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512].

FCS_IPSEC_EXT.1.5 The TSF shall implement the protocol: [

- *IKEv2 as defined in RFC 7296 and [with mandatory support for NAT traversal as specified in RFC 7296, section 2.23], 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-192, AES-CBC-256 (specified in RFC 3602), AES-GCM-128, AES-GCM-256 (specified in RFC 5282)].

FCS_IPSEC_EXT.1.7 The TSF shall ensure that [

- *IKEv2 SA lifetimes can be configured by a Security Administrator based on*

[

- *length of time, where the time values can be configured between [1 hour] and [8760 hours].*

]].

FCS_IPSEC_EXT.1.8 The TSF shall ensure that [

- *IKEv2 Child SA lifetimes can be configured by a Security Administrator based on*

[

- *number of bytes;*
- *length of time, where the time values can be configured between [1 hour] and [8760 hours].*

]].

TD0868 applied

FCS_IPSEC_EXT.1.9 The TSF shall generate the secret value x used in the IKE Diffie-Hellman key exchange (" x " in $g^x \bmod p$) using the random bit generator specified in FCS_RBG_EXT.1, and having a length of at least [225 (for DH Group 14), 256 (for DH Group 19), 384 (for DH Group 20)] bits.

- FCS_IPSEC_EXT.1.10** The TSF shall generate nonces used in *[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 IKE protocols implement DH Group(s) [
- *[14 (2048-bit MODP)] according to RFC 3526*
 - *[19 (256-bit Random ECP), 20 (384-bit Random ECP)] according to RFC 5114*
-].
- 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 *[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 *[IKEv2 CHILD_SA]* connection.
- FCS_IPSEC_EXT.1.13** The TSF shall ensure that *[IKEv2]* protocols perform peer authentication using a *[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 if the presented identifier in the received certificate matches the configured reference identifier, where the presented and reference identifiers are of the following fields and types: *[Distinguished Name (DN), SAN: IP address, SAN: Fully Qualified Domain Name (FQDN), SAN: user FQDN]*.

FCS_NTP_EXT.1 – NTP Protocol

- FCS_NTP_EXT.1.1** The TSF shall use only the following NTP version(s) [selection: NTP v4 (RFC 5905)].
- FCS_NTP_EXT.1.2** The TSF shall update its system time using [
- *Authentication using [SHA1, SHA256] as the message digest algorithm(s);*
- FCS_NTP_EXT.1.3** The TSF shall not update NTP timestamp from broadcast and/or multicast addresses.
- FCS_NTP_EXT.1.4** The TSF shall support configuration of at least three (3) NTP time sources in the Operational Environment.

FCS_SSH_EXT.1- SSH Protocol

- FCS_SSH_EXT.1.1** The TOE shall implement SSH acting as a *[server]* in accordance with that complies with RFCs 4251, 4252, 4253, 4254, *[4256, 4344, 5647, 5656, 6668, 8268, 8308, 8332]* and no other standard.
- FCS_SSH_EXT.1.2** The TSF shall ensure that the SSH protocol implementation supports the following authentication methods:
- *['password' (RFC 4252),*
 - *'publickey' (RFC 4252): [*
 - *rsa-sha2-256 (RFC 8332),*

- *rsa-sha2-512 (RFC 8332)]]*
- and no other methods.
- FCS_SSH_EXT.1.3** The TSF shall ensure that, as described in RFC 4253, packets greater than [262126 bytes] in an SSH transport connection are dropped.
- FCS_SSH_EXT.1.4:** The TSF shall protect data in transit from unauthorised disclosure using the following mechanisms:
- [
- *aes128-ctr (RFC 4344),*
 - *aes256-ctr (RFC 4344),*
 - *aes128-cbc (RFC 4253),*
 - *aes256-cbc (RFC 4253),*
 - *aes128-gcm@openssh.com (RFC 5647),*
 - *aes256-gcm@openssh.com (RFC 5647)]*
- and no other mechanisms.
- FCS_SSH_EXT.1.5:** The TSF shall protect data in transit from modification, deletion, and insertion using:
- [*hmac-sha2-256 (RFC 6668), hmac-sha2-512 (RFC 6668), implicit*] and no other mechanisms.
- FCS_SSH_EXT.1.6:** The TSF shall establish a shared secret with its peer using:
- [
- *diffie-hellman-group14-sha256 (RFC 8268),*
 - *diffie-hellman-group16-sha512 (RFC 8268),*
 - *diffie-hellman-group18-sha512 (RFC 8268),*
 - *ecdh-sha2-nistp256 (RFC 5656),*
 - *ecdh-sha2-nistp384 (RFC 5656),*
 - *ecdh-sha2-nistp521 (RFC 5656),]*
- and no other mechanisms.
- FCS_SSH_EXT.1.7:** The TSF shall use SSH KDF as defined in [*RFC 4253 (Section 7.2), RFC 5656 (Section 4)*] to derive the following cryptographic keys from a shared secret: session keys.
- FCS_SSH_EXT.1.8:** The TSF shall ensure that [*a rekey of the session keys*] occurs when any of the following thresholds are met:
- one hour connection time
 - no more than one gigabyte of transmitted data, or
 - no more than one gigabyte of received data.

FCS_SSHS_EXT.1 - SSH Protocol - Server

- FCS_SSHS_EXT.1.1:** The TSF shall authenticate itself to its peer (SSH Client) using:
- [
- *rsa-sha2-256 (RFC 8332),*
 - *rsa-sha2-512 (RFC 8332),*

].

FCS_TLSC_EXT.1 – TLS Client Protocol without Mutual Authentication – Controllers

- FCS_TLSC_EXT.1.1** The TSF shall implement [TLS 1.2 (RFC 5246), TLS 1.3 (RFC 8446)] supporting the following ciphersuites:
- [
- **Controllers (syslog channel)**
 - TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
 - TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289
 - TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
 - TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289
 - TLS_AES_128_GCM_SHA256
 - TLS_AES_256_GCM_SHA384
-] and no other ciphersuites.
- FCS_TLSC_EXT.1.2** The TSF shall verify that the presented identifier matches [*the reference identifier per RFC 6125 Section 6, IPv4 address in the CN or in the SAN*] and no other attribute types].
- FCS_TLSC_EXT.1.3** The TSF shall not establish a trusted channel if the server certificate is invalid [
- *without any administrator override mechanism*].
- FCS_TLSC_EXT.1.4** The TSF shall [*present the Supported Groups Extension with the following curves/groups: [secp256r1, secp384r1, secp521r1, ffdhe2048, ffdhe3072, ffdhe4096, ffdhe6144, ffdhe8192] and no other curves/groups*] in the Client Hello.
- FCS_TLSC_EXT.1.5** The TSF shall [
- *present the signature_algorithms extension with support for the following algorithms: [*
 - *rsa_pkcs1 with sha256(0x0401),*
 - *rsa_pkcs1with sha384(0x0501),*
 - *rsa_pkcs1 with sha512(0x0601),*
 - *ecdsa_secp256r1 with sha256(0x0403),*
 - *ecdsa_secp384r1 with sha384(0x0503),*
-] and no other algorithms;*
-].
- FCS_TLSC_EXT.1.6** The TSF [*does not provide*] the ability to configure the list of supported ciphersuites as defined in FCS_TLSC_EXT.1.1.
- FCS_TLSC_EXT.1.7** The TSF shall prohibit the use of the following extensions:
- Early data extension
 - Post-handshake client authentication according to RFC 8446, Section 4.2.6.
- FCS_TLSC_EXT.1.8** The TSF shall [*not use PSKs*].
- FCS_TLSC_EXT.1.9** The TSF shall [*reject [TLS 1.2, TLS 1.3] renegotiation attempts*].

FCS_TLSC_EXT.1 – TLS Client Protocol without Mutual Authentication – ION devices

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

- [
- **ION Devices (syslog channel)**
 - *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289*
 - *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289*
 - *TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
 - *TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
 - *TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
 - *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
 - *TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289*
 - *TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289*
 - **ION Devices (ITT channel)**
 - *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
-] and no other ciphersuites.
- FCS_TLSC_EXT.1.2** The TSF shall verify that the presented identifier matches [*the reference identifier per RFC 6125 Section 6, IPv4 address in the CN or in the SAN³*] and no other attribute types].
- FCS_TLSC_EXT.1.3** The TSF shall not establish a trusted channel if the server certificate is invalid [
- *without any administrator override mechanism*].
- FCS_TLSC_EXT.1.4** The TSF shall [*present the Supported Groups Extension with the following curves/groups: [secp256r1, secp384r1, secp521r1] and no other curves/groups*] in the Client Hello.
- FCS_TLSC_EXT.1.5** The TSF shall [
- *present the signature_algorithms extension with support for the following algorithms:* [
 - **ION Devices (Syslog channel)**
 - *rsa_pkcs1 with sha256(0x0401),*
 - *rsa_pkcs1with sha384(0x0501),*
 - *rsa_pkcs1 with sha512(0x0601),*
 - *ecdsa_secp256r1 with sha256(0x0403),*
 - *ecdsa_secp384r1 with sha384(0x0503),*
 - *ecdsa_secp521r1 with sha512(0x0603),*
 - **ION Devices (ITT channel)**
 - *rsa_pkcs1 with sha256(0x0401),*
 - *rsa_pkcs1with sha384(0x0501),*
 - *rsa_pkcs1 with sha512(0x0601),*
-] and no other algorithms;*
-].
- FCS_TLSC_EXT.1.6** The TSF [*does not provide*] the ability to configure the list of supported ciphersuites as defined in FCS_TLSC_EXT.1.1.
- FCS_TLSC_EXT.1.7** The TSF shall prohibit the use of the following extensions:

³ The ITT Channel only supports the reference identifier per RFC 6125 Section 6.

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

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

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

FCS_TLSC_EXT.2 – TLS Client Support for Mutual Authentication– ION devices ITT channel

FCS_TLSC_EXT.2.1: The TSF shall support TLS communication with mutual authentication using X.509v3 certificates.

FCS_TLSS_EXT.1 – TLS Server Protocol – Local WebUI (Controller)

FCS_TLSS_EXT.1.1 The TSF shall implement **[TLS 1.2 (RFC 5246), TLS 1.3 (RFC 8446)]** and reject all other TLS and SSL versions. The TLS implementation will support the following ciphersuites:

- [
- *TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288,*
 - *TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288*
 - *TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 8422,*
 - *TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 8422*
 - *TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
 - *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
 - *TLS_AES_128_GCM_SHA256*
 - *TLS_AES_256_GCM_SHA384*

] and no other ciphersuites.

FCS_TLSS_EXT.1.2 The TSF shall authenticate itself using X.509 certificate(s) using **[RSA with key size [2048, 3072, 4096] bits; ECDSA over NIST curves [secp256r1, secp384r1, secp521r1] and no other curves]**.

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

- *EC Diffie-Hellman key agreement over NIST curves [secp256r1, secp384r1, secp521r1] and no other curves;*
- *Diffie-Hellman parameters [of size 2048 bits, of size 3072 bits, of size 4096 bits]*

].

FCS_TLSS_EXT.1.4 The TSF shall support **[session resumption based on session tickets according to RFC 5077 (TLS 1.2), session resumption according to RFC 8446 (TLS 1.3)]**.

FCS_TLSS_EXT.1.5 The TSF **[does not provide]** the ability to configure the list of supported ciphersuites as defined in FCS_TLSS_EXT.1.1.

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

- Early data extension

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

FCS_TLSS_EXT.1.8 The TSF shall **[reject [TLS 1.2, TLS 1.3] renegotiation attempts]**.

FCS_TLSS_EXT.1 – TLS Server Protocol – Remote WebUI (Controller)

- FCS_TLSS_EXT.1.1** The TSF shall implement [**TLS 1.2 (RFC 5246)**] and reject all other TLS and SSL versions. The TLS implementation will support the following ciphersuites:
- [
- **TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288,**
 - **TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288**
 - **TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 8422,**
 - **TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 8422**
 - **TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289**
 - **TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289**
-] and no other ciphersuites.
- FCS_TLSS_EXT.1.2** The TSF shall authenticate itself using X.509 certificate(s) using [**RSA with key size [2048, 3072, 4096] bits**].
- FCS_TLSS_EXT.1.3** The TSF shall perform key exchange using: [
- **EC Diffie-Hellman key agreement over NIST curves [secp256r1, secp384r1, secp521r1] and no other curves;**
 - **Diffie-Hellman parameters [ffdhe6144, ffdhe8192]].**
-].
- FCS_TLSS_EXT.1.4** The TSF shall support [**session resumption based on session tickets according to RFC 5077 (TLS 1.2)**].
- FCS_TLSS_EXT.1.5** The TSF [**does not provide**] the ability to configure the list of supported ciphersuites as defined in FCS_TLSS_EXT.1.1.
- FCS_TLSS_EXT.1.6** The TSF shall prohibit the use of the following extensions:
- Early data extension
- FCS_TLSS_EXT.1.7** The TSF shall [**not use PSKs**].
- FCS_TLSS_EXT.1.8** The TSF shall [**reject [TLS 1.2] renegotiation attempts**].

FCS_TLSS_EXT.2 – TLS Server Support for Mutual Authentication – Remote WebUI (Controller)

- FCS_TLSS_EXT.2.1:** The TSF shall support TLS communication with mutual authentication of TLS clients using X.509v3 certificates and shall [**reject the connection if the client either does not provide a client certificate at all or the client certificate cannot be successfully validated by the TOE (except for override mechanisms that might be defined in FCS_TLSS_EXT.2.2) ('hard fail')**].
- FCS_TLSS_EXT.2.2:** When establishing a trusted channel, by default the TSF shall not establish a trusted channel if the client certificate is invalid. The TSF shall also [**not implement any administrator override mechanism**].
- FCS_TLSS_EXT.2.3:** The TSF shall not establish a trusted channel if the identifier contained in a certificate does not match an expected identifier for the client. If the identifier is a Fully Qualified Domain Name (FQDN), then the TSF shall match the identifiers according to RFC 6125, otherwise the TSF shall parse the identifier from the certificate and match the identifier against the expected identifier of the client as described in the TSS.
- FCS_TLSS_EXT.2.4:** The TSF shall present a [**TLS 1.2**] Certificate Request message containing the following algorithms: [selection:

FIA_UIA_EXT.1.3 The TSF shall provide the following remote authentication mechanisms [**Web GUI password, SSH password, SSH public key**] and [**no other mechanism**]. The TSF shall provide the following local authentication mechanisms [**password**].

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

TD0900 applied

FIA_UAU.7: Protected Authentication Feedback

FIA_UIA.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/ITT – X.509 Certificate Validation

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

- RFC 5280 certificate validation and certificate path validation **supporting a minimum path length of two certificates**.
- The certificate path must terminate with a trusted CA certificate designated as a trust anchor.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of the certificate using [**no revocation method**].
- The TSF shall validate the extendedKeyUsage field according to the following rules:
 - *Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.*
 - *Server certificates presented for DTLS/TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.*
 - *Client certificates presented for DTLS/TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the 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/ITT 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.1/Rev – X.509 Certificate Validation

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

- RFC 5280 certificate validation and certificate path validation **supporting a minimum path length of three certificates**.
- The certificate path must terminate with a trusted CA certificate designated as a trust anchor.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of the certificate using [**the Online Certificate Status Protocol (OCSP) as specified in RFC 6960**].

- The TSF shall validate the extendedKeyUsage field according to the following rules:
 - Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.
 - Server certificates presented for DTLS/TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.
 - Client certificates presented for DTLS/TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the 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/Rev The TSF shall only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.

FIA_X509_EXT.2 – X.509 Certificate Authentication

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

FIA_X509_EXT.2.2 When the TSF cannot establish a connection to determine the validity of a certificate, the TSF shall **reject the certificate**.

Application Note: For the syslog connection, the behavior is to not accept the server certificate and fail the connection (not configurable).

FIA_X509_EXT.3 – X.509 Certificate Requests

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

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

5.2.5 Security Management (FMT)

FMT_MOF.1/ManualUpdate – Management of Security Functions Behaviour

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

FMT_MOF.1/Services – Management of Security Functions Behaviour

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

FMT_MTD.1/CryptoKeys – Management of TSF Data

FMT_MTD.1.1/CryptoKeys The TSF shall restrict the ability to [[manage]] the [cryptographic keys **and certificates used for VPN operation**] to [Security Administrators].

FMT_MTD.1/CoreData – Management of TSF Data

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

FMT_SMF.1 – Specification of Management Functions

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

- *Ability to administer the TOE remotely;*
 - *Ability to configure the access banner;*
 - *Ability to configure the session inactivity time before session termination;*
 - *Ability to update the TOE, and to verify the updates using **digital signature** capability prior to installing those updates;*
- [
- *Ability to manage the cryptographic keys;*
 - *Ability to configure the cryptographic functionality;*
 - *Ability to configure thresholds for SSH rekeying;*
 - *Ability to configure the list of supported (D)TLS ciphers;*
 - *Ability to configure the interaction between TOE components;*
 - *Ability to configure the lifetime for IPsec SAs;*
 - *Ability to generate Certificate Signing Request (CSR) and process CA certificate response;*
 - *Ability to start and stop services;*
 - *Ability to configure local audit behaviour (e.g. changes to storage locations for audit; changes to behaviour when local audit storage space is full; changes to local audit storage size);*
 - *Ability to modify the behaviour of the transmission of audit data to an external IT entity;*
 - *Ability to configure the list of TOE-provided services available before an entity is identified and authenticated, as specified in FIA_UIA_EXT.1;*
 - *Ability to configure NTP;*
 - *Ability to configure the reference identifier for the peer;*
 - *Ability to manage the TOE's trust store and designate X509v3 certificates as trust anchors;*
 - *Ability to administer the TOE locally;*
 - *Ability to manage the trusted public keys database;*
 - *Ability to configure the authentication failure parameters for FIA_AFL.1*

].

TD0880 applied

FMT_SMR.2 – Restrictions on Security Roles

FMT_SMR.2.1 The TSF shall maintain the roles:

- *Security Administrator.*

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

FMT_SMR.2.3 The TSF shall ensure that the conditions

- *The Security Administrator role shall be able to administer the TOE remotely;*
- are satisfied.

5.2.6 Protection of the TSF (FPT)

FPT_SKP_EXT.1 – Protection of TSF data (for reading of all symmetric keys)

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

FPT_APW_EXT.1 – Protection of Administrator Passwords

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

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

FPT_STM_EXT.1 – Reliable Time Stamps

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

FPT_STM_EXT.1.2 The TSF shall *[synchronize time with an NTP server]*.

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 verify the integrity of the TOE firmware and software;*
- *Prior to providing any cryptographic service and [on-demand] to verify correct operation of cryptographic implementation necessary to fulfil the TSF;*
- *[start-up] self-tests [*
 - *AES ECB Encrypt Known Answer Test*
 - *AES ECB Decrypt Known Answer Test*
 - *AES GCM Encrypt Known Answer Test*
 - *AES GCM Decrypt Known Answer Test*
 - *AES CBC Encrypt Known Answer Test*
 - *AES CBC Decrypt Known Answer Test*
 - *RSA Sign Known Answer Test*
 - *RSA Verify Known Answer Test*
 - *ECDSA Sign Known Answer Test*
 - *ECDSA Verify Known Answer Test*
 - *HMAC-SHA-1 Known Answer Test*
 - *HMAC-SHA-256 Known Answer Test*
 - *HMAC-SHA-384 Known Answer Test*
 - *HMAC-SHA-512 Known Answer Test*
 - *SHA-1 Known Answer Test*
 - *DRBG SP800-90A Known Answer Tests*
 - *SP 800-90Arev1 Section 11.3 Health Tests*
 - *DH Known Answer Test*
 - *KAS-ECC-SSC Known Answer Test*
 - *TLS KDF Known Answer Tests*
 - *IKEv2 KDF Known Answer Tests*
 - *SSH KDF Known Answer Tests*

].

to demonstrate the correct operation of the TSF: noise source health tests.

TD0836 applied

FPT_TST_EXT.1.2 The TSF shall respond to [all failures] by [entering an error state called maintenance mode].

FPT_TUD_EXT.1 – Trusted Update

- FPT_TUD_EXT.1.1 The TSF shall provide *Security Administrators* the ability to query the currently executing version of the TOE firmware/software and **[no other TOE firmware/software version]**.
- FPT_TUD_EXT.1.2 The TSF shall provide *Security Administrators* the ability to manually initiate updates to TOE firmware/software and **[no other update mechanism]**.
- FPT_TUD_EXT.1.3 The TSF shall provide a means to authenticate firmware/software updates to the TOE using a digital signature mechanism and **[no other mechanisms]** prior to installing those updates.

FPT_ITT.1 – Basic Internal TSF Data Transfer Protection

- FPT_ITT.1.1 The TSF shall protect TSF data from disclosure and detect its modification when it is transmitted between separate parts of the TOE **through the use of [TLS]**.

5.2.7 TOE Access (FTA)

FTA_SSL.3 – TSF-initiated Termination

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

FTA_SSL.4 – User-initiated Termination

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

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_TAB.1 – Default TOE Access Banners

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

5.2.8 Trusted Path/Channels (FTP)

FTP_ITC.1 – Inter-TSF Trusted Channel

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

- FTP_ITC.1.2** The TSF shall permit [*the TSF, 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 using TLS,*
 - *Communicating with third party VPN*
-].

FTP_TRP.1/Admin – Trusted Path

- FTP_TRP.1.1/Admin** The TSF shall **be capable of using [SSH, HTTPS]** to provide a communication path between itself and **authorized remote Administrators ~~users~~** that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from disclosure **and provides detection of modification of the channel data.**
- FTP_TRP.1.2/Admin** The TSF shall permit remote Administrators ~~users~~ to initiate communication via the trusted path.
- FTP_TRP.1.3/Admin** The TSF shall require the use of the trusted path for initial Administrator authentication and all remote administrative actions.

5.3 TOE Security Assurance Requirements

The security assurance requirements for the TOE are included by reference to [NDcPP/SSH10].

Table 5 Assurance Components

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
	ALC_FLR.3: Systematic flaw remediation
ASE: Security Target Evaluation	ASE_INT.1: ST introduction
	ASE_CCL.1: Conformance claims
	ASE_SPD.1: Security problem definition
	ASE_OBJ.1: Security objectives for the operational environment
	ASE_ECD.1: Extended components definition
	ASE_REQ.1: Stated security requirements
	ASE_TSS.1: TOE summary specification
ATE: Tests	ATE_IND.1: Independent testing - conformance
AVA: Vulnerability Assessment	AVA_VAN.1: Vulnerability survey

Consequently, the evaluation activities specified in the following Supporting Documents apply to the TOE evaluation:

- Supporting Document Mandatory Technical Document: Evaluation Activities for Network Device cPP, December-6-2023, Version 3.0e

6. TOE Summary Specification

This chapter describes the security functions:

- Security Audit
- Communication
- Cryptographic Support
- Identification and Authentication
- Security Management
- Protection of the TSF
- TOE Access
- Trusted Path/Channels

6.1 Security Audit

FAU_GEN.1	The TOE is designed to be able to generate log records for security relevant and other events as they occur. The events that can cause an audit record to be logged include starting and stopping the audit function (also startup and shutdown of system), any use of an administrator functions via the Web Interface as well as all of the events identified in Table 4 (which corresponds to the audit events specified in the [NDcPP/SSH10]). All log records include the following contents: date/time, event type, user ID (i.e., username, IP address) or component (i.e., tls, syslog), and description of the event including success or failure. For user-initiated actions, the User ID is included in the log records. For cryptographic key operations, the key name—or certificate name if applicable—is also logged. Furthermore, based on the event, the description of the event will include additional information as required in Table 4. Refer to the CC AGD [CCECG] for the complete list of mandated audit logs and contents.
FAU_GEN.2	The TOE identifies the responsible user for each event based on the specific username and/or network entity (identified by source IP address) that caused the event.
FAU_STG_EXT.1 FAU_STG.1	The audit trail generated by the distributed TOE comprises several logs, which are locally stored in the TOE's component file system on the hard disk and viewed via the CLI or UI pages. The size of each log file is administrator configurable by specifying the percentage of space allocated to each log type on the hard disk (minimum 45 MB). If the log size is reduced, the TOE removes the oldest logs when the changes are committed. When a log reaches the maximum size, the TOE starts overwriting the oldest log entries with the new log entries. The distributed TOE stores the audit records locally on each component and protects them from unauthorized modification and deletion. The TOE does not provide an interface where a user can modify the audit records, thus it prevents modification to the audit records. The distributed TOE can be configured to send generated audit records to an external syslog server in real-time using TLS. Each TOE component (Controller and ION) can send their own audit records securely. When configured to send audit records to a syslog server, audit records are also written to the external syslog as they are written locally to the internal logs.

6.2 Communication

FCO_CPC_EXT.1	The TOE implements a registration process using a channel that meets the secure registration channel requirements in FTP_ITT.1. The TOE utilizes TLS to provide a registration channel between a new ION Device and the Controller. The TOE requires that an administrator first create a new ION Device object from within the Controller. The Controller then pushes the Controller's internal CA certificate to the ION Device, the ION Device creates a CSR that the Controllers' internal CA uses to issue the ION Device a TLS certificate, and the ION Device validates the received TLS certificate. The Controller then pushes a policy to the ION Device. After completion, the ION Device and Controller subsequently communicate through TLS connections. An administrator can disable communication between an ION Device and the Controller via the Controller GUI.
---------------	---

6.3 Cryptographic Support

FCS_CKM.1 FCS_CKM.2 FCS_COP.1 FCS_RBG_EXT.1	The TOE includes NIST-validated cryptographic algorithms provided by Palo Alto Networks SD-WAN Crypto Module supporting the cryptographic functions below. The following libraries have been tested: • Palo Alto Networks SD-WAN Instant-On Network (ION) Crypto Library – I, version 1.2 • Palo Alto Networks SD-WAN Virtual Instant-On Network (vION) Crypto Library – I, version 1.2 • Palo Alto Networks Controller Crypto Library II, version 1.1 The following functions have been certified in accordance with the identified standards. Table 6 Cryptographic Functions <table><tr><th>Functions</th><th>Standards</th><th>Certificates</th></tr><tr><td colspan="3">Asymmetric Key Generation and Asymmetric Key Generation/IKE (FCS_CKM.1)</td></tr><tr><td>ECC key pair generation (NIST curves P-256, P-384, P-521) <i>Note that TLS each use any of P-256, P-384, or P-521 for authentication, but certificate generation by the TOE for P-256 and P-384 key sizes only.</i></td><td>FIPS PUB 186-5</td><td>Appliances (#A6185): ECC RSA VMs (#A6238): ECC RSA</td></tr><tr><td>RSA key generation (key sizes 2048, 3072, 4096 bits)</td><td>FIPS PUB 186-5</td><td>Controller (#A8446): ECC RSA</td></tr><tr><td>FFC Schemes using “safe-prime” groups</td><td>RFC 3526 RFC7919 NIST SP 800-56Ar3</td><td>Tested with known good implementation</td></tr><tr><td colspan="3">Cryptographic Key Establishment (FCS_CKM.2)</td></tr></table>	Functions	Standards	Certificates	Asymmetric Key Generation and Asymmetric Key Generation/IKE (FCS_CKM.1)			ECC key pair generation (NIST curves P-256, P-384, P-521) <i>Note that TLS each use any of P-256, P-384, or P-521 for authentication, but certificate generation by the TOE for P-256 and P-384 key sizes only.</i>	FIPS PUB 186-5	Appliances (#A6185): ECC RSA VMs (#A6238): ECC RSA	RSA key generation (key sizes 2048, 3072, 4096 bits)	FIPS PUB 186-5	Controller (#A8446): ECC RSA	FFC Schemes using “safe-prime” groups	RFC 3526 RFC7919 NIST SP 800-56Ar3	Tested with known good implementation	Cryptographic Key Establishment (FCS_CKM.2)		
Functions	Standards	Certificates																	
Asymmetric Key Generation and Asymmetric Key Generation/IKE (FCS_CKM.1)																			
ECC key pair generation (NIST curves P-256, P-384, P-521) <i>Note that TLS each use any of P-256, P-384, or P-521 for authentication, but certificate generation by the TOE for P-256 and P-384 key sizes only.</i>	FIPS PUB 186-5	Appliances (#A6185): ECC RSA VMs (#A6238): ECC RSA																	
RSA key generation (key sizes 2048, 3072, 4096 bits)	FIPS PUB 186-5	Controller (#A8446): ECC RSA																	
FFC Schemes using “safe-prime” groups	RFC 3526 RFC7919 NIST SP 800-56Ar3	Tested with known good implementation																	
Cryptographic Key Establishment (FCS_CKM.2)																			

	RSA based key establishment (key sizes 2048, 3072, 4096 bits)	RSAES-PKCS1-v1_5 as specified in Section 7.2 of RFC 8017, "Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.2	Tested with known good implementation
	ECDSA based key establishment (NIST P-256, P-384, P-521)	NIST SP 800-56A	Appliances (#A6185): ECC VMs (#A6238): ECC Controller (#A8446): ECC
	FFC Schemes using "safe-prime" groups	RFC 3526 RFC7919 NIST SP 800-56Ar3	Tested with known good implementation
AES Data Encryption/Decryption (FCS_COP.1/DataEncryption)			
	AES CBC, CTR, GCM (128, 192, 256 bits) <i>The 192 bits AES key size is used in CBC mode only and for IPsec only.</i>	AES as specified in ISO 18033-3 CBC as specified in ISO 10116 CTR as specified in ISO 10116 GCM as specified in ISO 19772	Appliances (#A6185): AES VMs (#A6238): AES Controller (#A8446): AES
Signature Generation and Verification (FCS_COP.1/SigGen)			
	RSA Digital Signature Algorithm (rDSA) (modulus 2048, 3072, 4096)	FIPS PUB 186-5, "Digital Signature Standard (DSS)", Section 5.4, using PKCS #1 v2.2 Signature Schemes RSASSA-PSS and/or RSASSAPKCS1v1_5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3	Appliances (#A6185): RSA VMs (#A6238): RSA Controller (#A8446): RSA

	ECDSA (NIST curves P-256, P-384, and P-521)	FIPS PUB 186-5, "Digital Signature Standard (DSS)", Section 6 and Appendix D, Implementing "NIST curves" P-256, P-384, P-521 ISO/IEC 14888-3, Section 6.6	Appliances (#A6185): ECDSA VMs (#A6238): ECDSA Controller (#A8446): ECDSA
	Cryptographic Hashing (FCS_COP.1/Hash)		
	SHA-1, SHA-256, SHA-384 and SHA-512 (digest sizes 160, 256, 384 and 512 bits)	ISO/IEC 10118-3:2004	Appliances (#A6185): SHS VMs (#A6238): SHS Controller (#A8446): SHS
	Keyed-hash Message Authentication (FCS_COP.1/KeyedHash)		
	• 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)	ISO/IEC 9797-2:2011	Appliances (#A6185): HMAC VMs (#A6238): HMAC Controller (#A8446): HMAC
	Random Bit Generation (FCS_RBG_EXT.1)		
	CTR_DRBG (AES-256)	ISO/IEC 18031:2011	Appliances (#A6185): DRBG VMs (#A6238): DRBG Controller (#A8446): DRBG
The TOE implements the ISO/IEC 18031:2011 Deterministic Random Bit Generator (DRBG) based on the AES 256 block cipher in counter mode (CTR_DRBG(AES)). The TOE instantiates the DRBG with maximum security strength, obtaining the 256 bits of entropy to seed the DRBG. The hardware-based entropy source is described in the proprietary Entropy Design document. The TOE generates asymmetric cryptographic keys used for key establishment in accordance with FIPS PUB 186-5, "Digital Signature Standard (DSS)", Appendix B.3 for RSA schemes, FIPS PUB 186-5, "Digital Signature Standard (DSS)" and Appendix B.4 for ECC schemes, and NIST SP 800-56Ar3 and RFC 3526 for FFC schemes. In FIPS-CC mode, all non-Approved key generation and key			

establishment functions are disabled. Where required, only the CAVP-validated Approved key generation and key establishment functions are used for all protocols including SSH, TLS, HTTPS, IKE/IPsec.

While the TOE generally fulfills all of the FIPS PUB 186-5 requirements without extensions, the following table specifically identifies the “should”, “should not”, and “shall not” conditions from the publication along with an indication of whether the TOE conforms to those conditions with deviations rationalized. Key generation is among the identified sections.

Table 7 FIPS 186-5 Conformance

FIPS PUB 186-5	“should”, “should not”, or “shall not”	Implemented accordingly?	Rationale for deviation
FIPS PUB 186-5 Appendix B.3			
B.3.1	shall not	Yes	N/A
FIPS PUB 186-5 Appendix B.4			
B.4.1	should	Yes	N/A
B.4.2	should	Yes	N/A

The TOE performs cryptographic key establishment in accordance with NIST Special Publication 800-56A for elliptic curve-based key establishment⁴ schemes, and NIST Special Publication 800-56A for finite field-based key establishment⁵ schemes. The TOE acts as both a sender and as a recipient for all supported key establishment schemes (ECC, FFC, DH Group14). For TLS, the domain parameters used for the finite field-based key establishment scheme are compliance with FIPS 186-5. For IKE, the TOE uses safe primes as defined by RFC 3526⁶ section 3 key establishment scheme. Per SP 800-56Ar3, only the safe primes (e.g., MODP-2048) defined in Sections 3 of RFC 3526 can be used. RFC 7919 (FFDHE) is supported (FFDHE2048, FFDHE3072, FFDHE4096, FFDHE6144, FFDHE8192) for TLS clients.

The claimed key generation and key establishment algorithms used for each function is summarized below.

FCS_CKM.1:

- X.509 key pair generation: ECDSA (256, 384, 521), RSA (2048, 3072, 4096)
- TLS: RSA (2048, 3072, 4096), ECDSA (256, 384, 521), FFC (2048 or greater)
- IPsec: FFC DH Groups 14, ECDSA (256, 384)
- SSH: RSA (2048, 3072, 4096), ECDSA (256, 384, 521), FFC DH Groups 14, 16, 18

FCS_CKM.2:

- TLS: RSA (2048, 3072, 4096), ECC (256, 384, 521), FFC (2048 or greater)
- IKE/IPsec: FFC DH Groups 14, ECC (256, 384)⁷
- SSH: ECC (256, 384, 521), FFC DH Groups 14, 16, 18

The SHA-2 hash function is associated with the digital signature generation/verification and corresponding HMAC functions.

⁴ Elliptic curve-based key generation and establishment are used for HTTPS, TLS, and IKE connections.

⁵ Finite field-based key generation and establishment are used for HTTPS and TLS connections.

⁶ Diffie-Hellman Group 14 with safe primes based on RFC 3526 and SP 800-56Ar3 are used for IKE connections.

⁷ Corresponding to Groups 19, and 20.

FCS_CKM.4	Table 8 Private Keys and CSPs			
	CSP #	CSP/Key Name	Type	Description
	1	RSA Private Keys	RSA	RSA Private keys for verification of signatures or authentication (RSA 2048, 3072, or 4096-bit)
	2	ECDSA Private Keys	ECDSA	ECDSA Private key for verification of signatures and authentication (P-256, P-384, P-521)
	3	TLS Pre-Master Secret	TLS Secret	Secret value used to derive the TLS session keys
	4	TLS DHE/ECDHE Private Components	DH	Diffie-Hellman private FFC or EC component used in TLS (FFDHE 2048+, ECDHE P-256, P-384, P-521)
	5	TLS HMAC Keys	HMAC	TLS integrity and authentication session keys (HMAC-SHA2-256, HMAC-SHA2-384)
	6	TLS Encryption Keys	AES	TLS encryption session keys (128 and 256 CBC or 128 and 256 GCM)
	7	S-S VPN IPsec/IKE authentication Keys	HMAC	(SHA-256, SHA-384 or SHA- 512) Used to authenticate the peer in an IKE/IPsec tunnel connection.
	8	S-S VPN IPsec/IKE session Keys	AES	Used to encrypt IKE/IPsec data. These are AES (128, 192, 256 CBC and 128, 256 GCM) IKE keys and (128, 192, and 256 CBC, 128 and 256 GCM) IPsec keys
	9	S-S VPN IPsec/IKE Diffie Hellman Private Components	DH	Diffie-Hellman (Groups 14, 19, and 20) private component used in key establishment
	10	RA VPN IPsec session Keys	AES	(128, 192, and 256 CBC, 128 and 256 GCM) Used to encrypt remote access sessions utilizing IPsec.
	11	RA VPN IPsec authentication HMAC	HMAC	(SHA-256, SHA-384, SHA-512) Used in authentication of remote access IPsec data.
	12	Firmware code integrity check	HMAC ECDSA	Used to check the integrity of crypto-related code. (HMAC-SHA-256)
	13	Firmware Content Encryption Key	AES-256	Used to encrypt/decrypt firmware, software, and sensitive content.

	<table><tr><td>14</td><td>Password</td><td>Password</td><td>Authentication string with a minimum length of 8 characters. Stored hashed with SHA-256 and nonce.</td></tr><tr><td>15</td><td>DRBG Seed /State</td><td>DRBG</td><td>AES 256 CTR DRBG used in the generation of a random values.</td></tr></table> The TOE performs a key error detection check on each internal, intermediate transfer of a key. The TOE zeroizes (i.e., overwrite) non-persistent cryptographic keys as soon as their associated session has terminated. In addition, the TOE recognizes when a private key expires and promptly zeroizes the key on expiration. The TOE does not permit expired private signature keys to be archived. Private cryptographic keys, plaintext cryptographic keys, and all other critical security parameters stored in intermediate locations for the purposes of transferring the key/critical security parameters (CSPs) to another location are zeroized immediately following the transfer. Zeroization is done by overwriting the storage location with a random pattern, followed by a read-verify. Note that plaintext cryptographic keys and CSPs are only ever stored in volatile memory. For non-volatile memories other than EEPROM and Flash, the zeroization is executed by overwriting three times using a different alternating data pattern each time. This includes the SSD storage. This includes all CSPs that are not stored in volatile memory such as private keys, hashed passwords, and entropy seeds. For volatile memory and non-volatile EEPROM and Flash memories, the zeroization is executed by a single direct overwrite consisting of a pseudo random pattern generated by Approved DRBG. Sensitive data in volatile memory includes session keys such as encryption keys, integrity keys, pre-Master secret, etc.	14	Password	Password	Authentication string with a minimum length of 8 characters. Stored hashed with SHA-256 and nonce.	15	DRBG Seed /State	DRBG	AES 256 CTR DRBG used in the generation of a random values.
14	Password	Password	Authentication string with a minimum length of 8 characters. Stored hashed with SHA-256 and nonce.						
15	DRBG Seed /State	DRBG	AES 256 CTR DRBG used in the generation of a random values.						
FCS_HTTPS_EXT.1 FCS_TLSS_EXT.1 FCS_TLSS_EXT.2	The TOE provides a set of TLSS channels depicted in the following diagram TLSS <pre>graph TD TLSS --> Controller TLSS --> IONs Controller --> Port443 Controller --> Port9443 IONs --> NotApplicable Port443 --> TLSv12 Port9443 --> TLSv12_13 TLSv12 --> TLSS_EXT2 TLSS_EXT2 --> X509</pre> The TOE can be configured as a TLS server for certificate-based authentication for secure connections. The TOE uses TLS service profiles to specify a certificate and the allowed protocol versions for TLS services such as UI or TLS VPN. TLS PSK authentication is not supported by the TOE. The TOE provides TLS 1.2 on its remote server interface and TLS 1.2 and TLS 1.3 on its local server interface. The TOE’s HTTPS protocol complies with RFC 2818. The TOE supports mutual authentication when communicating with ION Devices for the internal communications channel. The ciphersuites are not configurable. The TOE provides the following ciphersuites on its two server interfaces: Local WebUI • TLS RSA WITH AES 128 GCM SHA256 as defined in RFC 5288.								

	• TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288 • TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 8422, • TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 8422 • TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 • TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 • TLS_AES_128_GCM_SHA256 • TLS_AES_256_GCM_SHA384 Remote (Web UI and ITT Channel) • TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288, • TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288 • TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 8422, • TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 8422 • TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 • TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 The TOE authenticates itself using: Local Web UI • ECDSA with curves secp256r1, secp384r1, and secp521r1 • RSA with key size of 2048, 3072, 4096 Remote (Web UI and ITT Channel) • RSA with key size of 2048, 3072, 4096 The key exchange parameters generated by the TOE are: Local Web UI • ECDHE implementing NIST curves secp256r1, secp384r1, and secp521r1 • Diffie-Hellman Ephemeral parameters with key size of 6144, 8192 bits MODP Remote (Web UI and ITT Channel) • ECDHE implementing NIST curves secp256r1, secp384r1, and secp521r1 • Diffie-Hellman Ephemeral parameters with key size of 6144, 8192 bits MODP The TLS Server ITT Channel interface presents certificate request message containing the following algorithms: • rsa_pkcs1 with sha256(0x0401) • rsa_pkcs1 with sha384(0x0501) • rsa_pkcs1 with sha512(0x0601) • rsa_pss_rsae with sha256(0x0804) • rsa_pss_rsae with sha384(0x0805) • rsa_pss_rsae with sha512(0x0806) The TOE denies connections from clients requesting connections using SSL 2.0, SSL 3.0, TLS 1.0, or TLS 1.1 and shall not establish a trusted channel if the fully qualified distinguished name (FQDN) in the subject or Subject Alternative Name (SAN) field contained in a certificate does not match the expected identifier for the peer. The TOE will match the FQDN identifier according to RFC 6125. The TOE also supports username as identifier for Web UI HTTPS connection. The TOE (as a TLS client) uses TLS to initiate a TLS connection to external syslog server. The TOE (as TLS server) receives inbound remote administration TLS traffic on the management (MGT) interface from TLS client (e.g., web browser). The key agreement parameters of the server key exchange message consist of the key establishment parameters generated by the TOE: Diffie-Hellman Ephemeral parameters with key size of 6144, 8192 bits MODP, ECDHE implementing NIST curves secp256r1, secp384r1, and secp521r1. The TOE supports session resumption
--	---

	using session tickets (TLS 1.2) for a single context (no configuration needed). The TOE checks if session tickets expire which would trigger a full handshake. The session tickets are encrypted with AES encryption and 128-bits encryption key plus 256-bits HMAC-SHA-256 key; and adhered to the structural format provided in section 4 of RFC 5077. Fallback password-based authentication is not supported for this interface. Session resumption for TLS 1.3 is supported according to RFC 8446.
FCS_TLSC_EXT.1	The TOE provides a set of TLSC channels depicted in the following diagram: <pre> graph TD TLSC --> Controller TLSC --> IONs Controller --> Syslog_Channel_Controller[Syslog Channel] Syslog_Channel_Controller --> TLSv12_13[TLSv1.2 / TLSv1.3] TLSv12_13 --> X509_Controller[X509] IONs --> Syslog_Channel_IONs[Syslog Channel] IONs --> ITT_Channel[ITT Channel] Syslog_Channel_IONs --> TLSv12_IONs[TLSv1.2] TLSv12_IONs --> X509_IONs[X509] ITT_Channel --> TLSv12_ITT[TLSv1.2] TLSv12_ITT --> TLSC_EXT2[TLSC_EXT.2] TLSC_EXT2 --> X509_ITT[X509] </pre> The ION Devices implement TLS v1.2 (RFC 5246) when acting as a client with the following non-configurable ciphersuites: Syslog channel • TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289 • TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289 • TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 • TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 • TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 • TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 • TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289 • TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289 ITT channel • TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 The ION Device TLSC connections support the following signature algorithms extensions: Syslog channel • rsa_pkcs1 with sha256(0x0401), • rsa_pkcs1with sha384(0x0501), • rsa_pkcs1 with sha512(0x0601), • ecda_secp256r1 with sha256(0x0403), • ecda_secp384r1 with sha384(0x0503), • ecda_secp521r1 with sha512(0x0603) ITT channel

	• rsa_pkcs1 with sha256(0x0401), • rsa_pkcs1with sha384(0x0501), • rsa_pkcs1 with sha512(0x0601) The ION Devices present the Supported Elliptic Curves/Supported Groups Extension in the Client Hello with the secp256r1, secp384r1, and secp521r1 NIST curves. The ION Devices supports the use of IPv4 addresses and FQDN as reference identifiers within a certificate's CommonName (CN) or Subject Alternate Name (SAN) extension for the syslog channel and just FQDN for the ITT channel. The TOE checks the SAN/CN when performing certificate validation. The ION Devices reject all renegotiation attempts. The ION Devices do not support wildcards or certificate pinning. The ITT channel supports mutual authentication for security when communicating with the Controller. The ciphersuites are not configurable The Controllers implement TLS v1.2 (RFC 5246) and TLS 1.3 (RFC 8446) when acting as a syslog client with the following non-configurable ciphersuites: • TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 • TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 • TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 • TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 • TLS_AES_128_GCM_SHA256 • TLS_AES_256_GCM_SHA384 The Controller TLSC connections support the following signature algorithms extensions: • rsa_pkcs1 with sha256(0x0401), • rsa_pkcs1with sha384(0x0501), • rsa_pkcs1 with sha512(0x0601), • ecdsa_secp256r1 with sha256(0x0403), • ecdsa_secp384r1 with sha384(0x0503), The Controller present the Supported Elliptic Curves/Supported Groups Extension in the Client Hello with the secp256r1, secp384r1, and secp521r1 NIST curves and Diffie-Hellman Ephemeral parameters with key size 2048, 3072, 4096, 6144, and 8192 bits MODP. The Controller supports the use of FQDN as reference identifiers within a certificate's CommonName (CN) or Subject Alternate Name (SAN) extension. The TOE checks the SAN/CN when performing certificate validation. The Controller rejects all renegotiation attempts. The Controller does not support wildcards or certificate pinning. The Controller does not use PSKs in TLS 1.3, including out of band PSKs.
FCS_IPSEC_EXT.1	The TOE provides mechanisms to implement an IPsec Security Policy Database (SPD) in tunnel mode 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 ACLs consist of policy rules and profiles. 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 (e.g., TCP, UDP), source IP address and destination IP address. The first rule that matches the traffic is applied. If a policy rule matching the traffic attributes is not found, or if it is found and it specifies a deny action, then the packet is dropped (or DISCARDED) and the session is deleted. If the application flow is allowed and no further security profiles are applied then it is forwarded (it is allowed to BYPASS the tunnel). If the application is allowed and there are additional security profiles set, it will be sent to the stream signature processor. The traffic matching the IPsec crypto Security profile would then flow through the IPsec tunnel and be classified as "PROTECTED". If there is no SA that the IPsec can use to protect this

	traffic to the peer, IPsec uses IKE to negotiate with the remote peer to set up the necessary IPsec SAs on behalf of the data flow. The negotiation uses information specified in the IKE Network Profiles. If the TOE receives a packet that does not match any rules in the SPD the TOE discards the packet. The TOE uses route-based VPNs where the firewall makes a routing decision based on the destination IP address. It is not necessary to define special rules or to make explicit reference to a VPN tunnel; routing and encryption decisions are determined only by the destination IP address. Packets matching the destination IP address are permitted otherwise they are denied. The TOE also supports Network Address Translation (NAT) policies (as defined in RFC 5996) where policies can be defined to specify whether source or destination IP addresses and ports are converted between public and private addresses and ports. For example, private source addresses can be translated to public addresses on traffic sent from an internal (trusted) zone to a public (untrusted) zone. NAT policy rules are based on the source and destination zones, the source and destination addresses, and the application service. The NAT policy rules are compared against the incoming traffic in sequence; the first rule that matches the incoming traffic is applied. If no rules match, then the flow is denied. The TOE supports IKEv2. IKEv2 SA lifetime limits can be configured by an authorized administrator and can be limited to 24 hours for phase 1 (range: 1 to 8760 hours) and 8 hours (range: 1 to 8760 hours) for phase 2 SAs. IKEv2 SA phase 2 lifetime limits can be established based on number of bytes. The IKEv2 (as defined in RFCs 5996 and 4868) protocols implemented by the TOE include DH Group 14 (2048-bit MODP), DH Groups 19 (256-bit Random ECP), and 20 (384-bit Random ECP), using peer X.509v3 certificate authentication conforming to RFC4945. The DH Group 14 is based on RFC 3526 (sections 3, 4, and 5) and ECP is based on elliptic curve in FCS_CKM.2. IKE keys used as part of the establishment are generated according to FIPS 186-4. 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 supports RSA for peer authentication as well as pre-shared keys. In addition, the TSF will only establish an IKE channel if the presented identifier in the peer X.509v3 certificate DN or SAN (Subject Alternative Name) field matches the configured identifier: Distinguished Name (DN), IP address (in SAN), Fully Qualified Domain Name (FQDN) in SAN, or user Fully Qualified Domain Name (user FQDN) in SAN. The TOE supports text based keys for Pre-shared keys. A TOE administrator specifies the PSK in the authentication configuration. The PSK includes a combination of upper and lower-case letters, numbers, and supported special characters and must be between 4-128 characters. The keys are generated using the AES-256-CTR Deterministic Random Bit Generator (DRBG), as specified in ISO/IEC 18031:2011, and the following corresponding security strength sizes (in bits) are used: 256 (for DH Group 14), 256 (for DH Group 19), and 256 (for DH Group 20) bits. The nonces used in the IKE exchanges are generated at least 128 bits in size and at least half the output size of the PRF hash. The TSF generates the nonces of length 256 (for DH Groups 14, 19, and 20) bits, and are generated with the Approved DRBG. The TSF generates the secret value x used in the IKE Diffie-Hellman key exchange ("x" in $g^x \bmod p$) using the random bit generator specified in FCS_RBG_EXT.1 and having a length of 224 (for DH Group 14), 256 (for DH Group 19), and 384 (for DH Group 20) bits. The TOE provides AES-CBC-128, AES-CBC-192, AES-CBC-256, AES-GCM-128, and AES-GCM-256 for encrypting IKEv2 payloads. The TOE provides AES-CBC-128, AES-
--	--

	CBC-192, AES-CBC-256, AES-GCM-128 and AES-GCM-256 for encrypting IPsec payloads. The administrator is instructed to ensure that the size of key used for ESP must be less than or equal to the key size used to protect the IKE payload. Once configured, the TSF will check that the symmetric key size used in the IKE is greater than or equal to the symmetric key size used in the ESP. The TOE implementation of IPsec protocol ESP complies with RFC 4303 using the algorithms specified in FCS_IPSEC_EXT.1.4 together with the HMAC specified in FCS_IPSEC_EXT.1.4. SHA-based HMAC function truncated output is not used.
FCS_NTP_EXT.1	The TOE supports NTPv4, authenticating the NTP server that it synchronizes to using a SHA1 or SHA256 message digest. The TOE allows one or more NTP servers to be configured. At least one is required for time synchronization to occur, but more than 3 NTP servers can be specified.
FCS_SSH_EXT.1 FCS_SSHS_EXT.1	The TOE supports SSHv2 server (compliant to RFCs 4251, 4252, 4253, 4254, 4256, 4344, 5647, 5656, 6668, 8268, 8308, 8332) with AES encryption/decryption algorithm (in CBC, CTR, or GCM mode [gcm@openssh.com]) with key sizes of 128 and 256 bits. No optional characteristics are supported. The TOE also supports HMAC-SHA-256, HMAC-SHA-512, implicit MAC (aes128-gcm@openssh.com and aes256-gcm@openssh.com) for integrity and authenticity. Both encryption and integrity algorithms are administrator-configurable but non-Approved algorithms such as 3DES, HMAC-MD5, diffie-hellman-group-1 are all disabled when FIPS-CC mode is enabled. Only the Approved encryption and integrity algorithms along with key exchange algorithms diffie-hellman-group14-sha256, diffie-hellman-group16-sha512, diffie-hellman-group18-sha512 (RFC 8268), ecdh-sha2-nistp256, ecdh-sha2-nistp384, and ecdh-sha2-nistp521 (based on elliptic curve from FCS_CKM.2) and authentication public-key algorithms rsa-sha2-256 and rsa-sha2-512 are permitted in the evaluated configuration. In addition, the TOE supports SSH KDF as specified in RFC 4253 when using FFC and RFC 5656 section 4 when using ECC. If the SSH client (in the operational environment) only support non-Approved algorithms, the SSH connection will be rejected by the TOE. SSH server supports RSA SSH host keys as specified in FCS_SSHS_EXT.1. The TOE uses OpenSSH implementation to support the SSHv2 connections. The TOE uses password and publickey as the authentication methods. For publickey, the administrator must associate the public key to the user. SSH packets are limited to 262,126 bytes and any packet over that size will be dropped (i.e., not processed farther and buffer containing the packet will be freed). The TOE manages a tracking mechanism for each SSH session so that it can initiate a new key exchange (rekey) when either a non-configurable amount of data (1024 MBs) or time (3600 seconds) has passed, whichever threshold occurs first.

6.4 Identification and Authentication

FIA_UIA_EXT.1 FIA_UAU.7	The TOE is designed to require users to be identified and authenticated before they can access any of the TOE functions. The only capabilities allowed prior to users authenticating are the display of the informative (login) banner and responding to ICMP request (e.g., ping or ICMP echo reply). The TOE maintains user accounts which it uses to control access to the TOE. When creating a new user account, the administrator specifies a username (i.e., user identity or ID), a password or X.509v3 certificate⁸/public-key, and a role. The
----------------------------	--

⁸ X.509 certificate is only supported for the UI.

	TOE uses the username and password attributes to identify and authenticate the user when the user logs in via the GUI. With public key-based authentication, a digital signature is exchanged and verified, in lieu of a password. The TOE does not echo passwords as they are entered, and the private keys are never transmitted. For UI, the default authentication method is password and for SSH, the authentication method is public-key. The administrators must configure public-key authentication which is supported for both HTTPS session. It uses the role attribute to specify user permissions and control what the user can do with the GUI. The administrator can logon to the GUI by using a secure connection (HTTPS) from a web browser. The TOE provides remote access to the GUI using HTTPS. Regardless of whether a user logs in using an HTTPS or SSH connection, a logon is successful when the username and password provided by the user matches a defined account on the TOE or when the username and digital signature is verified by the TOE (e.g., verify the possession of a private key that corresponds to the public key defined for that user account).
FIA_PMG_EXT.1	Passwords can be composed of upper and lower case letters, numbers and special characters ("!", "@", "#", "\$", "%", "^", "&", "*", "(", ")", ":", ";", "+", ",", "-", ".", "/", "\\", ":", ":", "<", "=", ">", "[", "\\", "]", " ", "`", "{", "}", and "~"). For any Admin, the password must be composed of at least one lower case, one upper case, and one number or special character. The minimum password length is configurable by the administrator from 12 up to 24 characters for the remote interface and 8 to 256 characters for the local interface.
FIA_AFL.1	The TOE logs all unsuccessful authentication attempts in the System Log and tracks the number of failed attempts via internal counters. The TOE can be configured to lock a user or authorized IT entity out after a configurable number (1 – 10; default 5) of unsuccessful authentication attempts. For the SSH interface, the lock can be configured to last a specified amount of time (1 – 1440 minutes; default 30 minutes) during which providing the correct credentials will still not allow access (i.e., locked out). For the remote HTTPS/TLS interface, the local administrator must login and select to unlock the account. In the rare situation where all administrators (customer created) are locked out at the same time, the Superuser role (predefined 'admin' account) with public-key authentication can be used to login. In addition, the user can also wait until the lockout time expires.
FIA_X509_EXT.1/Rev FIA_X509_EXT.1/ITT	The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for TLS (server authentication), IKE/IPsec, and HTTPS connections. Public key infrastructure (PKI) credentials, such as RSA keys and certificates are stored in the TOE's underlying file system on the appliance. Certificates and their associated private key are stored in a single container: the Certificate File. The PKCS#12 file consists of an Encrypted Private Key and X.509 Certificate. By default, all the private keys are protected since they are always stored in encrypted format using AES-256. The physical security of the appliance (A.PHYSICAL_PROTECTION) protects the appliance 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. Revocation checking is performed on the TLS and IPsec peer certificates presented to the TOE. The TOE supports Online Certificate Status Protocol (OCSP) verification for certificate profiles. No revocation checking is performed on internal TOE connections.
FIA_X509_EXT.2	

	The TOE uses the following rules for validating the extendedKeyUsage⁹ 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. • 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. The TOE validates a certificate path by ensuring the presence of the basicConstraints extension is present and the cA flag is set to TRUE for all CA certificates. The TOE forms a Certificate trust path by ensuring that the basic constraints are met, proper key usage parameters exist, the CA flag exists, performing a revocation check of each certificate in the path and performing the validity of the CA certificate. The TOE will not treat a certificate as a CA certificate if the basicConstraints extension is not present or the cA flag is not set to TRUE. The TOE supports certificate path validation for a minimum path length of three certificates and terminates with a trusted CA certificate (i.e., Root certificate). The TOE downloads and caches OCSP status information for every CA listed in the trusted CA list of the TOE. The OCSP status is cached for the 'next update time' that is configured on the OCSP responder. The TOE uses this received value as the cache time. OCSP responders can also be configured for other external devices if someone decides to use it. The TOE uses a hard coded 1 hour as next update time (cached time) in this case. Caching only applies to validated certificates; if a TOE never validated a certificate, the TOE cache does not store the OCSP information for the issuing CA. To use OCSP for verifying the revocation status of certificates, administrator must configure the TOE to access an OCSP responder (server). The entity that manages the OCSP responder can be a third-party certificate authority (CA) or, if the enterprise has its own PKI, the TOE itself. When the certificate status is unknown or cannot be determined, the session is denied.
FIA_X509_EXT.3	The authorized administrator may generate a certificate request as specified in RFC 2986 and provide the following information in the request: public key, Common Name, Organization, Organizational Unit, and Country. The administrator may also import a certificate and private key into the TOE from an enterprise certificate authority or obtain a certificate from an external CA. The TOE provides the ability for administrators to generate a Certificate Signing Request (CSR) with a multi-level organizational unit. When the administrators import a certificate based on the CSR, the TOE will validate the certificate chain is based on a trusted CA which must be present on the TOE. Otherwise, the TOE will reject the certificate and will not associate it with the CSR.

6.5 Security Management

FMT_MOF.1/ManualUpdate FMT_MOF.1/Services FMT_MTD.1/CoreData FMT_MTD.1/CryptoKeys	The TOE provides a GUI management interface as well as an CLI management interface. The GUI is accessible via secure connection to the management port remotely over HTTPS. The CLI is a remote SSH based interface. The restricted role-based privileges enable only authorized administrators to configure the TOE functions such as updating the TOE and manipulating TSF data. For examples, the ability to manage the TOE's trust store, enable or disable of start/stop services, configure audit behavior, and
--	--

⁹ Certificates are not used for trusted updates or executable code integrity.

	VPN keys/certificates are restricted to Security Administrators only. The Security Administrators can generate RSA and ECDSA keypair (public key embedded in certificate), import private key and certificate, and delete keys. The key management operations can be performed via the Web UI or CLI. The users must be identified and authenticated by the TOE prior to any access to the management functions (including those that manipulate the TSF data). All TOE components are subject to manual updates that can only be performed by an administrator.
FMT_SMF.1	The security management functions provided by the TOE include, but are not limited to: • Ability to administer the TOE remotely; • 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; • Ability to configure the authentication failure parameters for FIA_AFL.1; • Ability to configure the list of TOE-provided services available before an entity is identified and authenticated, as specified in FIA_UIA_EXT.1; • Ability to configure the cryptographic functionality; • Ability to configure thresholds for SSH rekeying; • Ability to configure the interaction between TOE components; • Ability to configure NTP; • Ability to import X.509v3 certificates to the TOE's trust store; • Ability to manage the TOE's trust store and designate X509v3 certificates as trust anchor; • Ability to configure the lifetime for IPsec SAs; • Ability to configure the reference identifier for the peer; • Ability to configure the IPsec functionality; • Ability to import X.509v3 certificates; • Ability to configure audit behavior (e.g. changes to storage locations for audit; changes to behaviour when local audit storage space is full); • Ability to modify the behaviour of the transmission of audit data to an external IT entity; • Ability to administer the TOE locally; • Ability to start and stop services; • Ability to manage the trusted public keys database.
FMT_SMR.2	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. By default, the TOE has the following pre-defined administrator roles: Admin, Network Admin, Read-Only, and Security Admin. These administrator roles (except Read-Only) are all considered Security Administrator as defined in the [NDcPP/SSH10] for the purposes of this ST. For example, a user with Admin role can create, modify, or delete user accounts but user with Read-Only role cannot. All roles (including Security Administrator as defined in the NDcPP/SSH10) can administer the TOE remotely via CLI or Web UI, and a user account can only be assigned one role at a time. The Security Administrator role shall be able to administer the TOE remotely and locally.

6.6 Protection of the TSF

FPT_SKP_EXT.1	Certificates and their associated private key are stored in a single container: the Certificate File. The PKCS#12 file consists of an Encrypted Private Key and X.509 Certificate. By default, all the private keys are protected since they are always stored in encrypted format using AES-256. The TOE prevents the reading of all keys by encrypting them with a Master Key using AES-256. The TOE does not provide an interface to read the Master Key. The TOE is designed specifically to prevent access to locally-stored cryptographically protected passwords and does not disclose any keys stored in the TOE. The TOE protects the confidentiality of user passwords by hashing the passwords using SHA-256. The TOE does not offer any functions that will disclose to any users a stored cryptographic key or password.
FPT_APW_EXT.1	
FPT_ITT.1	The distributed TOE communication between the Controllers and the ION Devices are all protected using TLS network connections. The communications are used for management purposes.
FPT_TST_EXT.1	The TOE meets FPT_TST_EXT requirements and therefore provides self-tests at start-up to demonstrate the correct operation of: key error detection, cryptographic algorithms, and DRBG. The self-tests also verify the integrity of stored TSF executable code and TSF data. The TOE performs the following Power-on self-tests: • AES ECB Encrypt Known Answer Test • AES ECB Decrypt Known Answer Test • AES GCM Encrypt Known Answer Test • AES GCM Decrypt Known Answer Test • AES CBC Encrypt Known Answer Test • AES CBC Decrypt Known Answer Test • RSA Sign Known Answer Test • RSA Verify Known Answer Test • ECDSA Sign Known Answer Test • ECDSA Verify Known Answer Test • HMAC-SHA-1 Known Answer Test • HMAC-SHA-256 Known Answer Test • HMAC-SHA-384 Known Answer Test • HMAC-SHA-512 Known Answer Test • SHA-1 Known Answer Test • DRBG SP800-90A Known Answer Tests • SP 800-90Arev1 Section 11.3 Health Tests • DH Known Answer Test • KAS-ECC-SSC Known Answer Test • TLS KDF Known Answer Tests • IKEv2 KDF Known Answer Test • SSH KDF Known Answer Test A known-answer test involves operating the cryptographic algorithm on data for which the correct output is already known and comparing the calculated output with the previously generated output (the known answer). If the calculated output does not equal the known answer, the known-answer test shall fail. The firmware integrity test verifies the signature of the image to ensure that it has not been modified or tampered with it. If the signature verification fails, the integrity test will fail. The entropy health tests consist of two tests, RCT (Repetition Count Test) and APT (Adaptive Proportion Test). RCT is designed to detect if the noise source become stuck outputting the same

	value too many times in a row. APT is designed to detect if the frequency of a sample value occurs too many times in a particular window size. The entropy health test will fail if any of the two tests fails. If a self-test (e.g., known answer test, entropy health tests, firmware integrity test using ECDSA as defined in FCS_COP.1/SigGen) fails, the TOE enters an error state and outputs an error indicator. The TOE doesn't perform any cryptographic operations or functions while in the error state. All data output from the TOE is inhibited when an error state exists. Should one or more power-up self-tests fail the module will reboot and enter maintenance mode (i.e., error state) in which the reason for the failure can be determined. In the maintenance mode, all operational and network functions will be unavailable with one exception which is the Reboot operation. These self-tests are sufficient to ensure correct functionality of the TSF because the self-tests encompass the cryptographic functionality and the integrity of the entire TOE software/firmware executable code.
FPT_TUD_EXT.1	Authorized administrators may query the current software/firmware version of the TOE. When updates are installed, the TOE need to be rebooted for the change to take place (no delayed activation). When updates are available from Palo Alto, an administrator can obtain and install those updates from updates.paloaltonetworks.com if there is an internet connection. For an additional layer of protection, Palo Alto Networks has chosen to sign (using RSA) all content that is downloaded to the TOE. If the TOE is not connected to the internet, the administrators can download the updates and upload it to the TOE. When the TOE update package and its corresponding digital signature is downloaded or uploaded; the digital signature is checked automatically by TOE by verifying the signature using the public key (corresponding to the RSA key used to create the signature). Palo Alto Networks manages the updates.paloaltonetworks.com and guarantees that images are digitally signed. Public keys are stored and protected on the TOE's file system. If the signature is verified, the update is performed; otherwise, the update is not performed.
FPT_STM_EXT.1	The TOE is a hardware appliance, or a virtual appliance image installed on a virtualization platform that includes a hardware-based real-time clock. The clock is based on a quartz crystal running at high frequency and accuracy. The hardware hosting the VM-Series provides the time clock, as well as CPU, ports, etc., which are provided by VM hypervisor. The TOE's embedded OS manages the clock and exposes administrator clock-related functions such as set time. Additionally, the TOE supports NTP. The clock is used for audit record time stamps, measuring session activity for termination, time-based lockout following authentication failure, IPsec session rekeying, and X.509 certificate validation.

6.7 TOE Access

FTA_SSL.3	The TOE will enforce an administrator-defined inactivity timeout value after which the remote session will be terminated regardless of authentication methods (e.g., password, public-key, X.509v3 certificate). The TOE can be configured by an administrator to set an interactive session timeout value (any integer value from 1 to 1,440 minutes with default set to 10 minutes). The function is enabled by default and the administrator must follow the CC AGD to configure the session idle timeout value. A remote session that is inactive (i.e., no commands issued from the remote client) for the defined timeout value will be terminated. The users will be required to re-enter their user ID and their password or perform public-key or certificate-based authentication, so they can establish a new session once a session is terminated. Users can also initiate termination of their own sessions. They can click on the "Logout" link or enter the "exit" command.
FTA_SSL.4	The TOE provides remote administrators the ability to logout (or terminate) their sessions as directed by the user.
FTA_SSL_EXT.1	The TOE terminates local administrator sessions after an Administrator-specified time period of inactivity.
FTA_TAB.1	The TOE can be configured to display an informative banner that will appear prior to authentication when accessing the TOE via remote connection to the management port in order to access the Web Interface (HTTPS) or CLI (SSH).

6.8 Trusted Path/Channels

FTP_ITC.1	The TOE can be configured to send audit records to external syslog server(s) using TLS or IPsec in real-time. Tunneling TLS connection over IPsec can be done using a service route for additional security. The TOE permits the TSF to initiate communication with the syslog server using TLS trusted channel. The TOE permits the following communications below: - Connecting with remote VPN gateways/peers using IPsec [TOE is peer], - Transmitting audit records to an audit server using TLS [TOE is client], The TOE communicates with its authorized trusted entities over TLS or IPsec, and all communication are sent over the trusted channel and are protected by the security protocols. The underlying cryptographic algorithms and implementation are CAVP-validated and are part of the TOE.
FTP_TRP.1/Admin	The TOE provides HTTPS (TLSv1.3 and TLSv1.2) and SSHv2 to support secure remote administration. Administrators can initiate a remote session that is secured (from disclosure and modification) using CAVP-validated cryptographic operations, and all remote security management functions require the use of a secure channel.

7. Protection Profile Claims

This ST is conformant to the [NDcPP/SSH10].

8. Rationale

This Security Target includes by reference the [NDcPP/SSH10] applicable Security Problem Definition, Security Objectives, and Security Assurance Requirements. The Security Target makes no additions to the [NDcPP/SSH10] assumptions. Security Functional Requirements have been reproduced verbatim with the Protection Profile and PP-Module operations completed except where refinements were made by the ST author and formatted per the defined convention. Operations on the security requirements follow [NDcPP/SSH10] application notes and evaluation activities. The Security Target did not add or remove any security requirements. Consequently, [NDcPP/SSH10] rationales apply and are complete.