Palo Alto Networks M-200, M-500, and M-600 Hardware, and Virtual Appliances all running Panorama 10.0.5

Version: 1.0
Date: June 21, 2021

Palo Alto Networks, Inc.
www.paloaltonetworks.com

Table of Contents

LIST OF FIGURES

LIST OF TABLES

1. Security Target Introduction

This section identifies the Security Target (ST) and Target of Evaluation (TOE) identification, ST conventions, ST conformance claims, and the ST organization. The TOE is a centralized management appliance running version 10.0.5, provided by Palo Alto Networks Inc. Panorama is available as a virtual or physical appliance, each of which supports licenses for managing up to 25, 100, or 1,000 next-generation firewalls.

The physical appliances include the M-200, M-500 and M-600 models and the virtual appliances include Panorama virtual appliances which are used to simplify central management and collect information on activity across all managed firewall and Wildfire appliances. Information can include network traffic, user activity, threats which allows for the TOE to centrally analyze, investigate, and report on the aggregated data.

The focus of this evaluation is on the TOE functionality supporting the claims in the collaborative Protection Profile for Network Devices. (See section 1.2 for specific version information).

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] includes protection of communications between the TOE and trusted external IT entities (trusted channel), protection of communications between the TOE and remote administrators (trusted path), identification and authentication of administrators, auditing of security-relevant events, ability to verify the source and integrity of updates to the TOE, implementation of session idle timeout, and the restricted use of FIPS Approved algorithms and protocols.

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 M-200, M-500, and M-600 Hardware, and Virtual Appliances all running Panorama 10.0.5 Security Target

ST Version – Version 1.0

ST Date – June 21, 2021

TOE Identification – Palo Alto Networks M-200, M-500, and M-600 Hardware, and Virtual Appliances all running Panorama 10.0.5.

The Panorama virtual appliance is supported on the following hypervisors:

- VMware
  - VMware ESXi with vSphere 5.5, 6.0, 6.5, or 6.7
- Microsoft Hyper-V Server 2012 R2
- Kernel-based Virtual Machine (KVM) on CentOS 7

The VM-Series virtual appliance must be the only guest running in the virtualized environment. Evaluation testing included the following:

VMware ESXi 6.5:

- Dell PowerEdge R730 Processor: Intel XEON CPU E5-2640 v4 (Broadwell microarchitecture) with Broadcom 5720 NIC
- Memory: 64 GB ECC DDR4 2133

Hyper-V Server 2012 R2 and KVM CentOS 7:

- Dell PowerEdge R730 Processor: Intel XEON CPU E5-2640 v4 (Broadwell microarchitecture) with Broadcom 5720 NIC
- Memory: 64 GB ECC DDR4 2133

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: This ST is conformant to:

- collaborative Protection Profile for Network Devices, Version 2.1, 24 September 2018 [NDcPP].

The following NIAP Technical Decisions apply to this PP and have been accounted for in the ST development and the conduct of the evaluation:

- 0484: NIT Technical Decision for Interactive sessions in FTA_SSL_EXT.1 & FTA_SSL.3
- 0483: NIT Technical Decision for Applicability of FPT_APW_EXT.1
- 0482: NIT Technical Decision for Identification of usage of cryptographic schemes
- 0481: NIT Technical Decision for FCS_(D)TLSC_EXT.X.2 IP addresses in reference identifiers
- 0480: NIT Technical Decision for Granularity of audit events
- 0478: NIT Technical Decision for Application Notes for FIA_X509_EXT.1
- 0477: NIT Technical Decision for Clarifying FPT_TUD_EXT.1 Trusted Update
- 0475: NIT Technical Decision for Separate traffic consideration for SSH rekey
- 0451: NIT Technical Decision for ITT Comm UUID Reference Identifier
- 0450: NIT Technical Decision for RSA-based ciphers and the Server Key Exchange message
- 0447: NIT Technical Decision for Using 'diffie-hellman-group-exchange-sha256' in FCS_SSHC/S_EXT.1.7
- 0425: NIT Technical Decision for Cut-and-paste Error for Guidance AA
- 0424: NIT Technical Decision for NDcPP v2.1 Clarification - FCS_SSHC/S_EXT1.5
- 0423: NIT Technical Decision for Clarification about application of RfI#201726rev2

- 0412: NIT Technical Decision for FCS_SSHS_EXT.1.5 SFR and AA discrepancy
- 0410: NIT technical decision for Redundant assurance activities associated with FAU_GEN.1
- 0409: NIT decision for Applicability of FIA_AFL.1 to key-based SSH authentication
- 0408: NIT Technical Decision for local vs. remote administrator accounts
- 0407: NIT Technical Decision for handling Certification of Cloud Deployments
- 0402: NIT Technical Decision for RSA-based FCS_CKM.2 Selection
- 0401: NIT Technical Decision for Reliance on external servers to meet SFRs
- 0400: NIT Technical Decision for FCS_CKM.2 and elliptic curve-based key establishment
- 0399: NIT Technical Decision for Manual installation of CRL (FIA_X509_EXT.2)
- 0398: NIT Technical Decision for FCS_SSH*EXT.1.1 RFCs for AES-CTR
- 0397: NIT Technical Decision for Fixing AES-CTR Mode Tests
- 0396: NIT Technical Decision for FCS_TLSC_EXT.1.1, Test 2
- 0395: NIT Technical Decision for Different Handling of TLS1.1 and TLS1.2

The following Technical Decisions have been issued against the [NDcPP], but are not applicable to the TOE.

- 0453: NIT Technical Decision for Clarify authentication methods for SSH clients can use to authenticate SSH se
- 0411: NIT Technical Decision for FCS_SSHC_EXT.1.5, Test 1 – Server and client side seem to be confused

- 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].
- 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].

- Selection/Assignment operations completed by the ST author was bolded to show that it was completed by the ST author and not taken as-is from the PP.

1.3.1 Terminology

The following terms and abbreviations are used in this ST:

Authentication Profile Define the authentication service that validates the login credentials of administrators when they access Panorama.

Device Group Group the managed firewalls into logical units. A device group enables grouping based on network segmentation, geographic location, organizational function, or any other common aspect of firewalls that require similar policy configurations.

Log Collector Aggregate logs from managed firewalls. When generating reports, Panorama queries the Log Collectors for log information, providing you visibility into all the network activity that your firewalls monitor.

PAN-DB A mode where the TOE functions as a URL filtering database as an on-premise appliance.

Role-Based Access Control Define the privileges and responsibilities of administrative users (administrators). Every administrator must have a user account that specifies a role and authentication method.

Security Profile A security profile specifies protection rules to apply when processing network traffic. The profiles supported by the TOE include the IPsec crypto Security profile, IKE Network profile, and Vulnerability profile.

Security Zone A grouping of TOE interfaces. Each TOE interface must be assigned to a zone before it can process traffic.

1.3.2 Acronyms

AES	Advanced Encryption Standard
CBC	Cipher-Block Chaining
CC	Common Criteria for Information Technology Security Evaluation
CEM	Common Evaluation Methodology for Information Technology Security
CM	Configuration Management
CLI	Command Line Interface
DH	Diffie-Hellman
DRBG	Deterministic Random Bit Generator
EEPROM	Electrically Erasable Programmable Read-Only Memory
FIPS	Federal Information Processing Standard
FSP	Functional Specification
FTP	File Transfer Protocol
GCM	Galois/Counter Mode
GUI	Graphical User Interface
HMAC	Hashed Message Authentication Code
HTTPS	Hypertext Transfer Protocol Secure
IP	Internet Protocol
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
IPsec	Internet Protocol Security
NIST	National Institute of Standards and Technology
PP	Protection Profile
REST	Representational State Transfer

RSA	Rivest, Shamir and Adleman (algorithm for public-key cryptography)
SA	Security Association
SAR	Security Assurance Requirement
SFR	Security Functional Requirement
SHA	Secure Hash Algorithm
SSH	Secure Shell
SSL	Secure Socket Layer
ST	Security Target
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Functions
URL	Uniform Resource Locator
VM	Virtual Machine
VPN	Virtual Private Network

2. Product Description

Palo Alto Networks Panorama management appliances provide centralized monitoring and management of Palo Alto Networks next-generation firewalls and Wildfire appliances. It provides a single location from which administrators can oversee all applications, users, and content traversing the whole network, and then use this knowledge to create application enablement policies that control and protect the network. Using Panorama for centralized policy and firewall management increases operational efficiency in managing and maintaining a network of firewalls.

This evaluation only includes the Panorama physical and virtual appliance models as identified in section 1.1. Palo Alto Networks next-generation firewalls were evaluated previously, and information about them and Wildfire are provided for completeness only.

The Palo Alto next-generation firewalls are network firewall appliances and virtual appliances on specified hardware used to manage enterprise network traffic flow using function-specific processing for networking, security, and management. The next-generation firewalls let the administrator specify security policies based on an accurate identification of each application seeking access to the protected network. The next-generation firewall uses packet inspection and a library of applications to distinguish between applications that have the same protocol and port, and to identify potentially malicious applications that use non-standard ports. The next-generation firewall also supports the establishment of Virtual Private Network (VPN) connections to other next-generation firewalls or third party security devices.

The WildFire appliance provides an on-premises WildFire private cloud, enabling the analysis of suspicious files in a sandbox environment without requiring the firewall to send files out of network. The WildFire appliance can be configured to host a WildFire private cloud where the firewall is configured to submit samples to the local WildFire appliance for analysis. The WildFire appliance sandboxes all files locally and analyzes them for malicious behaviors using the same engine the WildFire public cloud uses.

Panorama enables the administrator to effectively configure, manage, and monitor the firewalls and Wildfire appliances with central oversight. Even though firewall and Wildfire appliances can be managed locally, by using Panorama to manage the appliances, the following three major benefits can be achieved:

1. **Centralized Configuration and Deployment** - To simplify central management and rapid deployment of the firewalls and WildFire appliances on the network, use Panorama to pre-stage the firewalls and WildFire appliances for deployment. Administrators can then assemble the firewalls into groups, and create templates to apply a base network and device configuration and use device groups to administer globally shared and local policy rules.

2. **Aggregated Logging with Central Oversight for Analysis and Reporting** - Collect information on activity across all the managed firewalls on the network and centrally analyze, investigate and report on the data. This comprehensive view of network traffic, user activity, and the associated risks empowers the administrators to respond to potential threats using the rich set of policies to securely enable applications on the network.

3. **Delegation Administration** - Enables administrator to delegate or restrict access to global and local firewall configurations and policies.

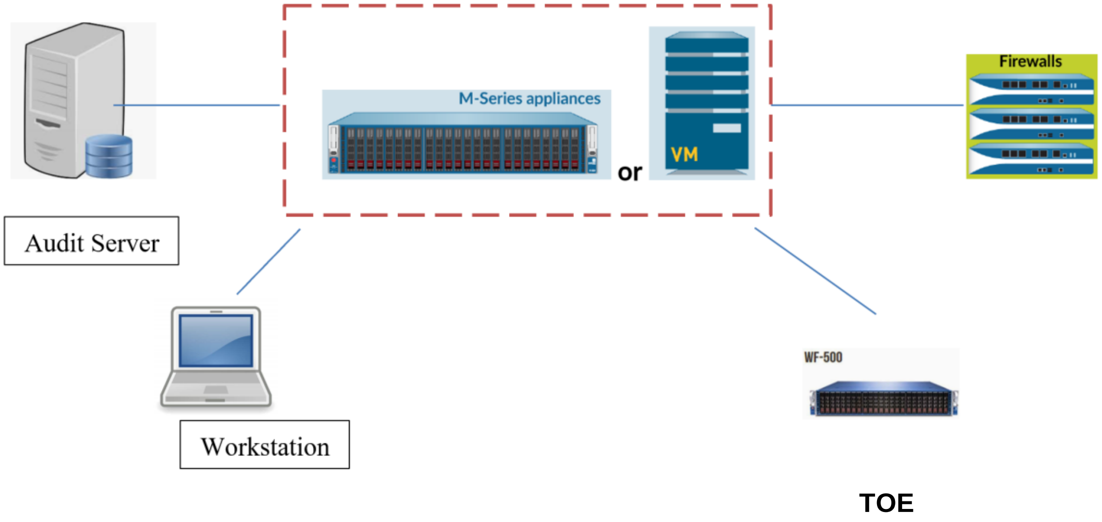

Figure 1: TOE Deployment

2.1 TOE Overview

The Target of Evaluation (TOE) is one or more Palo Alto Networks management appliance(s) that include Panorama M-200, M-500 and M-600 appliances and virtual appliances all running version 10.0.5. Panorama is available as a virtual or physical appliance, each of which supports licenses for managing up to 25, 100, or 1,000 firewalls. The TOE type is a network device as specified in the NDcPP.

In a deployment architecture, the Panorama security management appliance provides the capability to remotely manage multiple firewall appliances that control network traffic flow and WildFire appliances that analyze suspicious files traversing the network. However, since the firewall and Wildfire appliances are in the operational environment, these capabilities (i.e., stateful inspection filtering, IPsec VPN gateway, IPS/IDS threat prevention) are not evaluated (out of scope). Only the secure communication channels from Panorama to firewalls and Wildfires are claimed.

The administrators can deploy Panorama in the following system modes:

- Panorama - The appliance functions as a management server to manage firewalls and Dedicated Log Collectors. The appliance also supports a local Log Collector to aggregate firewall logs. This mode is the default mode.
- Management-Only - The appliance is a dedicated management appliance for your firewalls and Dedicated Log Collectors. The appliance has no firewall log collection capabilities.
- Log Collector - The appliance functions as a Dedicated Log Collector. In this deployment, the appliance has no web interface for administrative access, only a command line interface or CLI. When in Log Collector mode, Panorama is intended to be managed by another manager in Panorama or Management-Only mode (NOTE: this function is not evaluated). However, it provides a CLI with all of the administrative functions necessary to configure and manage the device.

Regardless of which system mode is deployed, the TOE must also be configured to run in the Common Criteria mode of operation described below. In addition, the TOE satisfies all mandatory SFRs regardless of system modes though some selection-based requirements such as HTTPS may not be relevant (for example, Log Collector mode does not support web interface).

Common Criteria Mode of Operation

The TOE is compliant with the capabilities outlined in this Security Target only when operated in Common Criteria mode (now referred to as FIPS-CC mode). FIPS-CC mode is a special operational mode in which the FIPS 140-2 requirements for startup and conditional self-tests as well as algorithm selection are enforced. In this mode, only CC Approved cryptographic algorithms and key sizes are available. All system modes with FIPS-CC mode enabled were tested in the evaluated configuration by the lab.

2.2 TOE Architecture

The TOE high-level architecture is divided into four main subsystems: system software (SS); database (DB); hardware (HW) and the hardened Linux-Derived operating system (OS). The system software provides system management functionality including proprietary software, management interfaces (CLI and GUI), cryptographic support (Palo Alto Networks Crypto Module), logging service (syslog-ng and auditd), web service (nginx), and authentication service. The database provides a data repository for audit logs, user account data, system data, configuration data, system log (i.e., syslog), and configuration logs. The operating system provides a customized Linux kernel to enforce domain separation, memory management, disk access, file I/O, network stacks (IPv4/IPv6), and communications with the underlying hardware components including the network interface cards (NICs), memory, CPUs, and hard disks. Only services and libraries required by the system software and DB are enabled in the OS. The virtual appliances will include the hypervisor as well (not shown in figure 2).

The following diagram depicts both the hardware and software architecture of the TOE.

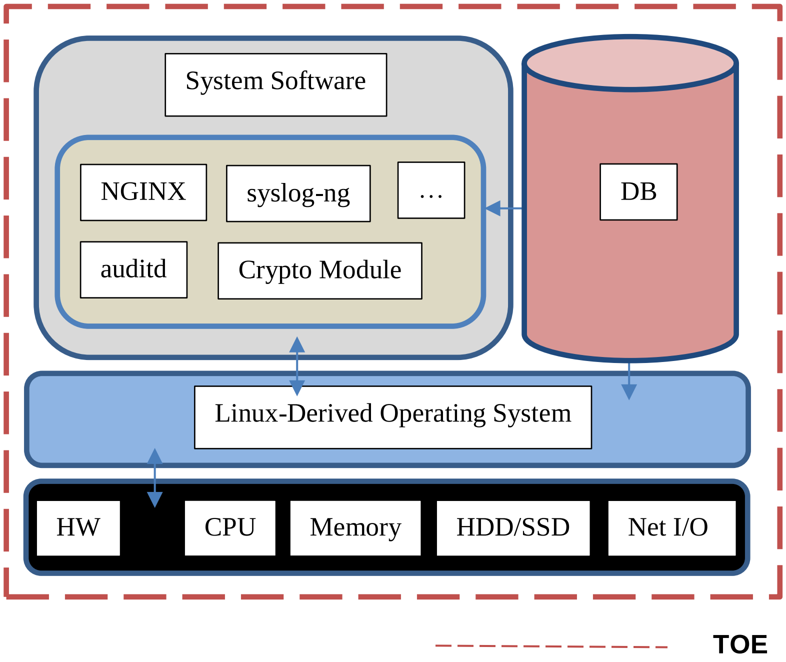

Figure 2: TOE Architecture

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 that provides the time stamp used for the audit records.
- Virtual appliances installed on specified hardware - the VM-Series supports the exact security functionalities available in the physical form factor appliances, allowing an administrator to safely enable physical or virtual appliances that enable applications flowing into, and across your virtual computing environments. The VM software and the appliances are both included in the TOE. The time clock, as well as CPU, ports, etc., are provided by VM environment (hypervisor) hosting the VMs. VMs are deployed in the system using Intel CPUs.

- Panorama OS software 10.0.5 – the software/firmware component that runs the appliance. For VMs, Panorama OS is software and for hardware appliances, Panorama OS is firmware. Panorama OS is built on top of a Linux kernel and runs along with NGINX (the web server that Palo Alto Networks uses), syslog-ng, sshd, Palo Alto Networks Crypto Module, and various vendor-developed applications that implement its capabilities.

The physical boundary of the TOE comprises the whole appliance (M-200, M-500, and M-600); and the virtual appliances on specified hypervisor and hardware. The models only differ in their performance capability (e.g., processor speed, memory, and disk space), but they all provide the same security functionality.

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

- M-200: 3 RJ-45 10Mbps/100Mbps/1000Mbps ports for network/management traffic (Ethernet ports); 1 RJ-45 10Mbps/100Mbps/1000Mbps port to access the device GUI through an Ethernet interface (management port); and 1 console port for connecting a serial console (management console port).
- M-500: 4 RJ-45 10Mbps/100Mbps/1000Mbps ports for network/management traffic (Ethernet ports); 2 10 GigE ports for network/management traffic (Gigabit Ethernet ports); 1 RJ-45 10Mbps/100Mbps/1000Mbps port to access the device GUI through an Ethernet interface (management port); and 1 console port for connecting a serial console (management console port).
- M-600: 3 RJ-45 10Mbps/100Mbps/1000Mbps ports for network/management traffic (Ethernet ports); 2 10 GigE ports for network/management traffic (Gigabit Ethernet ports); 1 RJ-45 10Mbps/100Mbps/1000Mbps port to access the device GUI through an Ethernet interface (management port); and 1 console port for connecting a serial console (management console port).

In the evaluated configuration, the TOE can be managed by:

- A computer either directly connected or remotely connected to the Management port via an RJ-45 Ethernet cable. The Management port is an out-of-band management port that provides access to the GUI/API via HTTPS or CLI via SSH. The computer is part of the operational environment and required to have a web browser (for accessing the GUI) and SSH client (for accessing the CLI).

System logs, which record information about the system such as authentication attempts, session idle timeout, and sessions establishment, termination, failures, are logged and stored locally by default. Configuration logs, which record all management actions are also logged and stored locally by default.

Table 1 TOE Platforms

Product Identification	Illustration	Description
M-200		Processor: Intel Xeon E5-2620 Memory: 128 GB DDR4 Maximum Logging Rate as Manager: Undisclosed Maximum Log Storage on Appliance: 16 TB (4 8TB RAID disks) SSD Storage Space: 240 GB

Product Identification	Illustration	Description
M-500		Processor: Intel Xeon E5-2637 Memory: 128 GB DDR3 Maximum Logging Rate as Manager: 20,000 logs per second Maximum Log Storage on Appliance: 24 TB (24 2TB or 1TB RAID disks) SSD Storage Space: 240 GB
M-600		Processor: Intel Xeon E5-2680 Memory: 256 GB DDR4 Maximum Logging Rate as Manager: Undisclosed Maximum Log Storage on Appliance: 48 TB (12 8TB RAID disks) SSD Storage Space: 240 GB
Virtual Appliances		
On VMware ESXi		Processor: See section 1.1. Memory: Up to 64 GB (min 16 GB) Maximum Logging Rate as Manager: 10,000 logs per second Maximum Log Storage on Appliance: 24 TB (12 virtual logging disks) SSD Storage Space: N/A
On Hyper-V		Processor: See section 1.1. Memory: Up to 32 GB (min 8 GB)

Product Identification	Illustration	Description
		Maximum Logging Rate as Manager: 10,000 logs per second Maximum Log Storage on Appliance: 24 TB (12 virtual logging disks) SSD Storage Space: N/A
On KVM		Processor: See section 1.1. Memory: Up to 32 GB (min 8 GB) Maximum Logging Rate as Manager: 10,000 logs per second Maximum Log Storage on Appliance: 24 TB (12 virtual logging disks) SSD Storage Space: N/A

The operational environment includes the following:

- Syslog server
- Palo Alto Networks Firewall or Wildfire appliances
- Workstation
  - Web browsers - Internet Edge (Release 42 or later), Firefox (version 66.0.5 or later), Safari (version 12.0.3 or later on Mac, and version 5.1.7 or later on Windows and iOS), and Chrome (version 74 or later) browser.
  - SSHv2 client

2.2.2 Logical Boundaries

This section summarizes the security functions provided by the TOE:

- Security Audit
- 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 for security relevant events including the events specified in [NDcPP]. By default, the TOE stores the logs locally so they can be accessed by an administrator. The TOE can also be configured to send the logs securely to a designated external log server.

2.2.2.2 Cryptographic Support

The TOE implements NIST-validated cryptographic algorithms that provide key management, random bit generation (RBG), encryption/decryption, digital signature generation and verification, cryptographic hashing, and keyed-hash message authentication features in support of higher level cryptographic protocols, including SSH 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 consistent with the FIPS 140-2 standard and [NDcPP]. All physical and virtual appliance included in the TOE are CAVP-validated and details are provided in the TOE Security Summary (TSS):

- The M-Series appliance are covered by CAVP certificates (#C1005).
  - AES - [FCS_COP.1/DataEncryption]
  - RSA, ECDSA, DSA - [FCS_COP.1/SigGen and FCS_CKM.1]
  - SHS - [FCS_COP.1/Hash]
  - HMAC - [FCS_COP/1/KeyedHash]
  - DRBG - [FCS_RBG_EXT.1]
  - Component - [FCS_CKM.2]
- The VM-Series virtual appliances are covered by CAVP certificates (#C999).
  - AES - [FCS_COP.1/DataEncryption]
  - RSA, ECDSA, DSA - [FCS_COP.1/SigGen and FCS_CKM.1]
  - SHS - [FCS_COP.1/Hash]
  - HMAC - [FCS_COP/1/KeyedHash]
  - DRBG - [FCS_RBG_EXT.1]
  - Component - [FCS_CKM.2]

2.2.2.3 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 offers network accessible (HTTPS, SSH) and direct connections to the GUI and SSH for interactive administrator sessions and HTTPS for XML and REST APIs.

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 human user and to associate that user with an authorized role. In addition, the TOE can authenticate users using X509 certificates and can be configured to lock a user out after a configurable number of unsuccessful authentication attempts.

2.2.2.4 Security Management

The TOE provides a GUI, CLI, or API (XML and REST) 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/CLI locally via direct RJ-45 Ethernet cable connection and remotely using an HTTPS/TLS or SSHv2 client.

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 audit function, configure the idle timeout, and review the audit trail. The TOE provides pre-defined Security Administrator, Audit Administrator, and Cryptographic Administrator roles. These administrator roles are all considered Security Administrator as defined in the [NDcPP] for the purposes of this ST.

2.2.2.5 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 its own timing mechanism 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. It also includes mechanism to verify TOE updates to prevent malicious or other unexpected changes in the TOE.

2.2.2.6 TOE Access

The TOE provides the capabilities for both TOE- and user-initiated locking of interactive sessions and for TOE termination of an interactive session after a period of inactivity. The TOE will display an advisory and consent warning message regarding unauthorized use of the TOE before establishing a user session.

2.2.2.7 Trusted Path/Channels

The TOE protects interactive communication with remote administrators using SSH or HTTP over TLS (HTTPS). SSH and TLS ensure both integrity and disclosure protection.

The TOE protects communication with the syslog server, Palo Alto Networks firewalls and Wildfire Appliances using TLS connections.

2.3 TOE Documentation

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

For Panorama 10.0.5, these documents include:

- Palo Alto Networks Common Criteria Evaluated Configuration Guide (CCECG) for Panorama 10.0.5, June 21, 2021
- VM-Series 10.0 Deployment Guide, Last Revised: See Link Below

  https://docs.paloaltonetworks.com/vm-series/10-0/vm-series-deployment.html
- PAN-OS® and Panorama 10.0 API Guide, Last Revised: See Link Below

  https://docs.paloaltonetworks.com/pan-os/10-0/pan-os-panorama-api/get-started-with-the-pan-os-rest-api/access-the-rest-api.html
- Panorama Administrator’s Guide Version 10-0, Last Revised: See Link Below

  https://docs.paloaltonetworks.com/content/dam/techdocs/en_US/pdf/panorama/10-0/panorama-admin/panorama-admin.pdf

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, SSH. The features below are out of scope.

Table 2 Excluded Features

Feature	Description
Telnet and HTTP Management Protocols	Telnet and HTTP are disabled by default and cannot be enabled in the evaluated configuration. Telnet and HTTP are insecure protocols which allow for plaintext passwords to be transmitted. Use SSH and HTTPS only as the management protocols to manage the TOE.
External Authentication Servers	The NDcPP does not require external authentication servers.
Shell and Console Access	The shell and console access is only allowed for pre-operational installation, configuration, and post-operational maintenance and trouble shooting.
API request over HTTP	By default, the TOE support API requests over HTTPS only. API request over HTTP is disabled and cannot be enabled in the evaluated configuration.
Stateful inspection filtering, VPN gateway, IPS/IDS threat prevention, URL filtering (PAN-DB), Log forwarding, and Malware sandboxing.	These features are provided by Palo Alto Networks firewalls and Wildfire appliances and are not included in this evaluation. Only the secure TLS connections between the Panorama to the firewalls and/or Wildfire to the TOE were evaluated.
Centralized Device Management.	These features (e.g., Policy Template and Push, Device Group) were not evaluated. Only the secure TLS connections between the firewalls and Wildfire to the TOE were evaluated.
SD-WAN	The PAN-OS software can include a native SD-WAN subscription to provide intelligent and dynamic path selection on top of what the PAN-OS security software already delivers. Secure SD-WAN provides the optimal end user experience by leveraging multiple ISP links to ensure application performance and scale capacity. The SD-WAN capability is considered out of scope for the Panorama evaluation.
Scheduled Reports	Reports can be configured to run immediately or schedule them to run at specific intervals. The TOE can save and export the reports or email them to specific recipients. The Scheduled Reports capability has not been tested and is considered out of scope for the Panorama evaluation.

Feature	Description
Automatic Panorama Connection Recovery	To ensure that you do not commit a configuration change that inadvertently causes the firewall to lose connectivity to Panorama, PAN-OS 9.1 can automatically revert the Panorama and firewall configuration to the previous running configuration. For example, if you perform configuration changes to the service routes, and as a result the change blocks traffic from the firewall to Panorama, the firewall's hourly connectivity checks can trigger Automatic Panorama Connection Recovery to revert the configuration back to the last running configuration to restore the connection to Panorama. This recovery ensures that a configuration change won't cause a loss in productivity or require you to physically access the firewall. The Automatic Panorama Connection Recovery was not evaluated. Only the secure TLS connections between the firewalls and Wildfire to the TOE were evaluated.
SAML Authentication	SAML Authentication is an XML-based open-standard for transferring identity data between two parties: an identity provider (IdP) and a service provider (SP). External authentication is outside the scope of the evaluation.
Any features not associated with SFRs in claimed NDcPP	NDcPP forbids adding additional requirements to the Security Target (ST). If additional functionalities 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].

In general, the [NDcPP] 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]. 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] 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.

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.

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]:

The SARs are the set of SARs specified in [NDcPP].

5.1 Extended Requirements

All of the extended requirements in this ST have been drawn from the [NDcPP]. The [NDcPP] defines all the extended SFRs (*_EXT.1) and since they are not redefined in this ST, the [NDcPP] should be consulted for more information in regard to 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
FAU: Security Audit	FAU_GEN.1: Audit Data Generation
	FAU_GEN.2: User Identity Association
	FAU_STG_EXT.1: Protected Audit Event Storage
FCS: Cryptographic Support	FCS_CKM.1: Cryptographic Key Generation
	FCS_CKM.2: Cryptographic Key Establishment
	FCS_CKM.4: Cryptographic Key Destruction
	FCS_COP.1/DataEncryption: Cryptographic Operation (AES Data Encryption/Decryption)
	FCS_COP.1/SigGen: Cryptographic Operation (Signature Generation and Verification)
	FCS_COP.1/Hash: Cryptographic Operation (Hash Algorithm)
	FCS_COP.1/KeyedHash: Cryptographic Operation (Keyed Hash Algorithm)
	FCS_RBG_EXT.1: Random Bit Generation
	FCS_HTTPS_EXT.1: HTTPS Protocol
	FCS_SSHS_EXT.1: SSH Server Protocol
	FCS_TLSC_EXT.1: TLS Client Protocol
	FCS_TLSC_EXT.2: TLS Client Protocol with Authentication
	FCS_TLSS_EXT.1: TLS Server Protocol
	FCS_TLSS_EXT.2: TLS Server Protocol with Mutual Authentication
FIA: Identification and Authentication	FIA_AFL.1: Authentication Failure Management
	FIA_PMG_EXT.1: Password Management
	FIA_UIA_EXT.1: User Identification and Authentication

Requirement Class	Requirement Component
	FIA_UAU_EXT.2: Password-based Authentication Mechanism
	FIA_UAU.7: Protected Authentication Feedback
	FIA_X509_EXT.1/Rev: X.509 Certificate Validation
	FIA_X509_EXT.2: X.509 Certificate Authentication
	FIA_X509_EXT.3: X.509 Certificate Requests
FMT: Security Management	FMT_MOF.1/ManualUpdate: Management of Security Functions Behaviour
	FMT_MTD.1/CoreData: Management of TSF Data
	FMT_SMF.1: Specification of Management Functions
	FMT_SMR.2: Restrictions on Security Roles
FPT: Protection of the TSF	FPT_SKP_EXT.1: Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)
	FPT_APW_EXT.1: Protection of Administrator Passwords
	FPT_TST_EXT.1: TSF Testing
	FPT_TUD_EXT.1: Trusted Update
	FPT_STM_EXT.1: Reliable Time Stamps
FTA: TOE Access	FTA_SSL_EXT.1: TSF-initiated Session Locking
	FTA_SSL.3: TSF-initiated Termination
	FTA_SSL.4: User-initiated Termination
	FTA_TAB.1: Default TOE Access Banners
FTP: Trusted Path/Channels	FTP_ITC.1: Inter-TSF Trusted channel
	FTP_TRP.1/Admin: Trusted Path

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:

a) Start-up and shutdown of the audit functions;

b) All auditable events for the <u>not specified</u> level of audit; and

c) *All administrative actions comprising:*
- *Administrative login and logout (name of user account shall be logged if individual user 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).*
- *[**no other actions**];*

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, 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	None.	None.
FCS_CKM.1	None.	None.
FCS_CKM.2	None.	None.
FCS_CKM.4	None.	None.
FCS_COP.1/DataEncryption	None.	None.
FCS_COP.1/SigGen	None.	None.
FCS_COP.1/Hash	None.	None.
FCS_COP.1/KeyedHash	None.	None.
FCS_RBG_EXT.1	None.	None
FCS_HTTPS_EXT.1/*	Failure to establish an HTTPS session.	Reason for failure
FCS_SSHS_EXT.1	Failure to establish a SSH session.	Reason for failure.
FCS_TLSC_EXT.1	Failure to establish a TLS session.	Reason for failure
FCS_TLSC_EXT.2	Failure to establish a TLS session.	Reason for failure
FCS_TLSS_EXT.1	Failure to establish a TLS session.	Reason for failure
FCS_TLSS_EXT.2	Failure to establish a TLS session.	Reason for failure
FIA_AFL.1	Unsuccessful login attempts limit is met or exceeded.	Origin of the attempt (e.g., IP address)
FIA_PMG_EXT.1	None.	None.
FIA_UIA_EXT.1	All use of identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UAU_EXT.2	All use of identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UAU.7	None.	None.
FIA_X509_EXT.1/Rev	Unsuccessful attempt to validate a certificate Any addition, replacement or removal of trust anchors[1] in the TOE's trust store	Reason for failure of certificate validation Identification of certificates added, replaced or removed as

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

Requirement	Auditable Events	Additional Audit Record Contents
		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_MTD.1/CoreData	None.	None.
FMT_SMF.1	All management activities of TSF data.	None.
FMT_SMR.2	None.	None.
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_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_EXT.1 (if “terminate the session” is selected)	The termination of a local session by the session locking mechanism.	None.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	None.
FTA_SSL.4	The termination of an interactive session.	None.
FTA_TAB.1	None.	None.
FTP_ITC.1	Initiation of the trusted channel. Termination of the trusted channel. Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempt.
FTP_TRP.1/Admin	Initiation of the trusted path. Termination of the trusted path. Failures of the trusted path functions.	None.

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 [

- ***TOE shall consist of a single standalone component that stores audit data locally***].

FAU_STG_EXT.1.3 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.

5.2.2 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-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3;***
- ***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;***
- ***FFC schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.1***
- ***FFC Schemes using Diffie-Hellman group 14 that meet the following: RFC 3526, Section 3***

] ~~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 3447, "Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1;***
- ***Elliptic curve-based key establishment schemes that meets the following: NIST Special Publication 800-56A Revision 2, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography";***
- ***Finite field-based key establishment schemes that meets the following: NIST Special Publication 800-56A Revision 2,***

"Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography"

- ***Key establishment scheme using Diffie-Hellman group 14 that meets the following RFC 3526, Section 3***

] ~~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 [[a 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* [***CBC, CTR, GCM***] *mode* and cryptographic key sizes [***128 bits, 192 bits, 256 bits***] that meet the following: *AES as specified in ISO 18033-3*, ***[CBC as specified in ISO 10116, CTR as specified in ISO 10116, GCM as specified in ISO 19772]***.

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 and cryptographic key sizes (modulus) [2048 bits, 3072 bits],***
- ***Elliptic Curve Digital Signature Algorithm and cryptographic key sizes [256 bits, 384 bits, 521 bits]***

] ~~and cryptographic key sizes [assignment: cryptographic key sizes]~~

that meet the following: *[*

- ***For RSA schemes: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSAPKCS1v1_5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3,***
- ***For ECDSA schemes: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 6 and Appendix D, Implementing "NIST curves" [P-256, P-384, P-521]; ISO/IEC 14888-3, Section 6.4***

].

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-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512***] and cryptographic key sizes [***160, 256, 384, 512***] **and message digest sizes [*160, 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] hardware-based noise source***] with 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 using TLS.

FCS_HTTPS_EXT.1.3 If a peer certificate is presented, the TSF shall [***not require client authentication, not establish the connection***] if the peer certificate is deemed invalid.

Application Note*: By default, the TOE acting as a HTTPS server does not perform mutual authentication for HTTPS client/user. If the TOE is configured for mutual authentication, the TLS client/user certificate must be valid or the TOE will not establish a HTTPS session (second selection).*

Application Note*: The TOE in Log Collector system mode does not claim HTTPS as the web interface is disabled in this mode.*

FCS_SSHS_EXT.1 – SSH Server Protocol

FCS_SSHS_EXT.1.1 The TSF shall implement the SSH protocol that complies with RFC(s) [***4251, 4252, 4253, 4254, 4344, 5656, 6668***].

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

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

FCS_SSHS_EXT.1.4 The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms and rejects all other encryption algorithms: [***aes128cbc,***

aes256-cbc, aes128-ctr, aes256-ctr, aes128-gcm@openssh.com, aes256-gcm@openssh.com].

FCS_SSHS_EXT.1.5 The TSF shall ensure that the SSH public-key based authentication implementation uses [***ssh-rsa***] as its public key algorithm(s) and rejects all other public key algorithms.

FCS_SSHS_EXT.1.6 The TSF shall ensure that the SSH transport implementation uses [***hmac-sha1, hmac-sha2-256, hmac-sha2-512***, ***implicit***] as its MAC algorithm(s) and rejects all other MAC algorithm(s).

FCS_SSHS_EXT.1.7 The TSF shall ensure that [***diffie-hellman-group14-sha1, ecdh-sha2-nistp256***] and [***ecdh-sha2-nistp384, ecdh-sha2-nistp521***] are the only allowed key exchange methods used for the SSH protocol.

FCS_SSHS_EXT.1.8 The TSF shall ensure that within SSH connections, the same session keys are used for a threshold of no longer than one hour, and each encryption key is used to protect no more than one gigabyte of data. After any of the thresholds are reached, rekey needs to be performed.

Application Note*: FCS_SSHS_EXT.1.8 has been modified by TD0475.*

FCS_TLSC_EXT.1 - TLS Client Protocol

FCS_TLSC_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_CBC_SHA as defined in RFC 3268***
- ***TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268***
- ***TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268***
- ***TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268***
- ***TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492***
- ***TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492***
- ***TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492***
- ***TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492***
- ***TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246***
- ***TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246***
- ***TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246***
- ***TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246***
- ***TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288***
- ***TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288***
- ***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***

].

FCS_TLSC_EXT.1.2 The TSF shall verify that the presented identifiers of the following types: [**identifiers defined in RFC 6125, IPv4 address in CN or SAN, IPv6 address in the CN or SAN**] are matched to reference identifiers.

Application Note*: FCS_TLSC_EXT.1.2 has been modified by TD0481.*

FCS_TLSC_EXT.1.3 When establishing a trusted channel, by default the TSF shall not establish a trusted channel if the server certificate is invalid. The TSF shall also [

- ***Not implement any administrator override mechanism***].

FCS_TLSC_EXT.1.4 The TSF shall [***present the Supported Elliptic Curves Extension with the following NIST curves: [secp256r1, secp384r1, secp521r1] and no other curves***] in the Client Hello.

Application Note*: The TOE connecting to the syslog server acts as a TOE client.*

FCS_TLSC_EXT.2 - TLS Client Protocol with authentication

FCS_TLSC_EXT.2.1 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_CBC_SHA as defined in RFC 3268***
- ***TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268***
- ***TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268***
- ***TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268***
- ***TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492***
- ***TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492***
- ***TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492***
- ***TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492***
- ***TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246***
- ***TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246***
- ***TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256 as defined in RFC 5246***

- ***TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246***
- ***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***

].

FCS_TLSC_EXT.2.2 The TSF shall verify that the presented identifiers of the following types: [**identifiers defined in RFC 6125, IPv4 address in CN or SAN, IPv6 address in the CN or SAN**] are matched to reference identifiers.

Application Note*: FCS_TLSC_EXT.2.2 has been modified by TD0481.*

FCS_TLSC_EXT.2.3 When establishing a trusted channel, by default the TSF shall not establish a trusted channel if the server certificate is invalid. The TSF shall also [

- ***Not implement any administrator override mechanism***].

FCS_TLSC_EXT.2.4 The TSF shall [***present the Supported Elliptic Curves Extension with the following NIST curves: [secp256r1, secp384r1, secp521r1] and no other curves***] in the Client Hello.

FCS_TLSC_EXT.2.5 The TSF shall support mutual authentication using X.509v3 certificates.

Application Note*: The TOE optionally supports mutual authentication for the secure syslog server connection.*

FCS_TLSS_EXT.1 - TLS Server Protocol

FCS_TLSS_EXT.1.1 The TSF shall implement [***TLS 1.2 (RFC 5246), TLS 1.1 (RFC 4346)***] and reject all other TLS and SSL versions. The TLS implementation will support the following ciphersuites:

[

- ***TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268***
- ***TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268***
- ***TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492***

- ***TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492***
- ***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***

].

FCS_TLSS_EXT.1.2 The TSF shall deny connections from clients requesting SSL 2.0, SSL 3.0, TLS 1.0, and **[*none*]**.

FCS_TLSS_EXT.1.3 The TSF shall [***perform RSA key establishment with key size [2048 bits, 3072 bits]; generate EC Diffie-Hellman parameters over NIST curves [secp256r1, secp384r1] and no other curves; generate Diffie-Hellman parameters of size [2048 bits]***].

Application Note*: For the management connection, the TOE is the TLS server. Mutual authentication is supported but must be configured.*

FCS_TLSS_EXT.2 - TLS Server Protocol with mutual authentication

FCS_TLSS_EXT.2.1 The TSF shall implement [***TLS 1.2 (RFC 5246), TLS 1.1 (RFC 4346)***] and reject all other TLS and SSL versions. The TLS implementation will support the following ciphersuites:

[

- ***TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268***
- ***TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268***
- ***TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492***
- ***TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492***
- ***TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492***
- ***TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492***
- ***TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256 as defined in RFC 5246***
- ***TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246***
- ***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***

].

FCS_TLSS_EXT.2.2 The TSF shall deny connections from clients requesting SSL 2.0, SSL 3.0, TLS 1.0, and [***none***].

FCS_TLSS_EXT.2.3 The TSF shall [***perform RSA key establishment with key size [2048 bits, 3072 bits]; generate EC Diffie-Hellman parameters over NIST curves [secp256r1, secp384r1] and no other curves; generate Diffie-Hellman parameters of size [2048 bits]***].

FCS_TLSS_EXT.2.4 The TSF shall support mutual authentication of TLS clients using X.509v3 certificates.

FCS_TLSS_EXT.2.5 When establishing a trusted channel, by default the TSF shall not establish a trusted channel if the server certificate is invalid. The TSF shall also [

- ***Not implement any administrator override mechanism***].

FCS_TLSS_EXT.2.6 The TSF shall not establish a trusted channel if the distinguished name (DN) or Subject Alternative Name (SAN) contained in a certificate does not match the expected identifier for the client.

Application Note*: Mutual authentication is supported for the management session (optional) and TOE to firewall/Wildfire connections (required).*

5.2.3 Identification and Authentication (FIA)

FIA_AFL.1 – Authentication Failure Management

FIA_AFL.1.1 The TSF shall detect when an Administrator configurable positive integer within [***1-10***] unsuccessful authentication attempts occur related to *Administrators attempting to authenticate remotely using a password*.

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

FIA_PMG_EXT.1 – Password Management

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

1. Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters: [***“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(”, “)”, [““”, “+”, “,”, “-”, “.”, “/”, “:”, “;”, “<”, “=”, “>”, “[”, “\”, “]”, “_”, “`”, “{”, “}”, and “~”]***];
2. Minimum password length shall be configurable to between [***6***] and [***15***] characters.

FIA_UIA_EXT.1 – User Identification and Authentication

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

- Display the warning banner in accordance with FTA_TAB.1;
- **[*[ICMP]*]**.

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

FIA_UAU_EXT.2 – Password-based Authentication Mechanism

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

FIA_UAU.7 – Protected Authentication Feedback

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

FIA_X509_EXT.1/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 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 [**<u>the Online Certificate Status Protocol (OCSP) as specified in RFC 6960, a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3, Certificate Revocation List (CRL) as specified in RFC 5759 Section 5</u>**].
- The TSF shall validate the extendedKeyUsage field according to the following rules:
  - *Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.*
  - *Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.*
  - *Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.*
  - *OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.*

FIA_X509_EXT.1.2/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 **[*TLS, HTTPS*],** 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 [***not accept the certificate***].

Application Note*: For the syslog connection, the behavior is not accept the server certificate and fail the connection. For the connection to firewall or Wildfire, the behavior is not accept the client certificate and fail the connection.*

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 **[<u>Common Name, Organization, Organizational Unit, Country</u>]**.

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

5.2.4 Security Management (FMT)

FMT_MOF.1/ManualUpdate - Management of Security Functions Behaviour

FMT_MOF.1.1/ManualUpdate The TSF shall restrict the ability to <u>enable</u> the functions *to perform manual update to Security Administrators*.

FMT_MTD.1/CoreData – Management of TSF Data

FMT_MTD.1.1/CoreData The TSF shall restrict the ability to *<u>manage</u>* 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 locally and 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 set the time which is used for time-stamps;***
- ***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***

].

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

are satisfied.

5.2.5 Protection of the TSF (FPT)

FPT_SKP_EXT.1 – Protection of TSF data (for reading of all pre-shared, symmetric and private keys)

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

Application Note*: FPT_APW_EXT.1 has been modified by TD0483.*

FPT_TST_EXT.1 – TSF Testing

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

- ***AES Encrypt Known Answer Test***
- ***AES Decrypt Known Answer Test***
- ***AES GCM Encrypt Known Answer Test***
- ***AES GCM Decrypt Known Answer Test***
- ***AES CCM Encrypt Known Answer Test***
- ***AES CCM Decrypt Known Answer Test***
- ***RSA Sign Known Answer Test***
- ***RSA Verify Known Answer Test***
- ***RSA Encrypt/Decrypt 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***
- ***SHA-256 Known Answer Test***
- ***SHA-384 Known Answer Test***
- ***SHA-512 Known Answer Test***
- ***DRBG SP800-90A Known Answer Tests***
- ***SP 800-90A Section 11.3 Health Tests***
- ***DH Known Answer Test***
- ***ECDH Known Answer Test***

- ***Firmware Integrity Test***

]

FPT_TUD_EXT.1 – Trusted Update

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

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

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

FPT_STM_EXT.1 – Reliable Time Stamps

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

FPT_STM_EXT.1.2 The TSF shall [***allow the Security Administrator to set the time***].

5.2.6 TOE Access (FTA)

FTA_SSL_EXT.1 – TSF-initiated Session Locking

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

FTA_SSL.3 – TSF-initiated Termination

FTA_SSL.3.1 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 **Administrator**-initiated termination of the **Administrator's** own interactive session.

FTA_TAB.1 – Default TOE Access Banners

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

5.2.7 Trusted Path/Channels (FTP)

FTP_ITC.1 – Inter-TSF Trusted Channel

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

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

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for *[*

- ***transmitting audit records to an audit server using TLS***
- ***communicating with Palo Alto Networks firewall or Wildfire appliances****]*.

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** 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** 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].

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

- Supporting Document Mandatory Technical Document: Evaluation Activities for Network Device cPP, September-2018, Version 2.1

6. TOE Summary Specification

This chapter describes the security functions:

- Security Audit
- 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 command via the Web Interface or CLI, as well as all of the events identified in Table 4 (which corresponds to the audit events specified in the [NDcPP]. All log records include the following contents: date/time, event type, user ID (i.e., username, IP address) or component (i.e., ssh, 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 the key is embedded in certificate or certificate request—is also logged. Furthermore, based on the event, the description of the event will include additional information as required in Table 4. Please 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	The audit trail generated by the TOE comprises several logs, which are locally stored in the TOE file system on the hard disk: • Configuration logs—include events such as when an administrator configures the security policies, user management, cryptographic functions, audit functions (e.g., enable syslog over TLS connection), and when an administrator configures which events gets audited. • System logs—include events such as user login and logout, session establishment, termination, and failures. The size of each log file is administrator configurable by specifying the percentage of space allocated to each log type on the hard disk. 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. Maximum disk space is platform dependent and it depends on the hard disk drive installed on the system. By default, the TOE allocates 25% to system log and 30% to configuration log. On a M-200, that is 12.48 GB and 14.98 GB, respectively. On VM, it's 4% each with about 633 MB allocated for each log type but this will depend on the size of the virtual disk allocated. The TOE stores the audit records locally and protects them from unauthorized deletion by allowing only users in the pre-defined Audit Administrator role to access the audit trail with delete privileges. The pre-defined Audit Administrator role is part of the Security Administrator role as defined by the [NDcPP]. The TOE does not provide an interface

	where a user can modify the audit records, thus it prevents modification to the audit records. The TOE can be configured to send generated audit records to an external Syslog server in real-time using TLSv1.2. 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 Cryptographic Support

FCS_CKM.1 FCS_CKM.2 FCS_COP.1/* FCS_RBG_EXT.1	The TOE includes NIST-validated cryptographic algorithms providing supporting cryptographic functions. The following functions have been certified in accordance with the identified standards.

Table 6 Cryptographic Functions

Functions	Standards	Certificates
Asymmetric Key Generation		
FFC key pair generation (key size 2048 bits)	FIPS PUB 186-4	**Appliances:** DSA #C1005 ECDSA # C1005 RSA # C1005 **VMs:** DSA #C999 ECDSA # C999 RSA # C999
ECC key pair generation (NIST curves P-256, P-384, P-521)	FIPS PUB 186-4	
RSA key generation (key sizes 2048, 3072 bits)	FIPS PUB 186-4	
Cryptographic Key Establishment		
RSA based key establishment	RSAES-PKCS1-v1_5	RSA = N/A **Appliances:** Component #C1005 **VMs:** Component #C999
ECDSA based key establishment	NIST SP 800-56A	
FFC based key establishment	NIST SP 800-56A	
AES Data Encryption/Decryption		

	AES CBC, CTR, GCM (128, 192, 256 bits)	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:** AES #C1005 **VMs:** AES #C999
	Signature Generation and Verification		
	RSA Digital Signature Algorithm (rDSA) (modulus 2048, 3072)	FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSAPKCS1v1_5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3	**Appliances:** RSA #C1005 **VMs:** RSA #C999
	ECDSA (NIST curves P-256, P-384, and P-521)	FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Section 6 and Appendix D, Implementing “NIST curves” P-256, P-384, ISO/IEC 14888-3, Section 6.4	**Appliances:** ECDSA #C1005 **VMs:** ECDSA #C999
	Cryptographic Hashing		
	SHA-1, SHA-256, SHA-384 and SHA-512 (digest sizes 160, 256, 384 and 512 bits)	ISO/IEC 10118-3:2004	**Appliances:** SHS #C1005 **VMs:** SHS #C999
	Keyed-hash Message Authentication		

• HMAC-SHA-1 (block size 512 bits, key size 160 bits and digest size 160 bits) • 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:** HMAC #C1005 **VMs:** HMAC #C999
Random Bit Generation		
CTR_DRBG (AES) from a hardware-based noise source with one independent software-based noise source of 256 bits of non-determinism	ISO/IEC 18031:2011	**Appliances:** DRBG #C1005 **VMs:** DRBG #C999

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 186-4, “Digital Signature Standard (DSS)”, Appendix B.3 for RSA schemes, FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.4 for ECC schemes, and FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.1 for FFC schemes.

While the TOE generally fulfills all of the FIPS PUB 186-4 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 186-4 Conformance

FIPS PUB 186-4	“should”, “should not”, or “shall not”	Implemented accordingly?	Rationale for deviation
FIPS PUB 186-4 Appendix B.1			
B.1.1	should	Yes	N/A
B.1.2	should	Yes	N/A
FIPS PUB 186-4 Appendix B.3			
B.3.1	shall not	Yes	N/A

FIPS PUB 186-4 Appendix B.4			
B.4.1	should	Yes	N/A
B.4.2	should	Yes	N/A

The TOE performs cryptographic RSA-based key establishment in accordance with RSAES-PKCS1-v1_5 as specified in Section 7.2 of RFC 3447, "Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1", NIST Special Publication 800-56A for elliptic curve-based key establishment schemes, and NIST Special Publication 800-56A for finite field-based key establishment schemes. The TOE acts as both a sender and as a recipient for all supported key establishment schemes (RSA, ECC, FFC). The TOE does not reveal specific details about an error (e.g., decryption error) for RSA-based key establishment schemes. For TLS, the domain parameters used for the finite field-based key establishment scheme are compliance with FIPS 186-4. For SSH, the TOE uses Diffie-Hellman Group 14 that meets RFC 3526 section 3 key establishment scheme as specified in RFC 4253 section 6.5.

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, authentication or key establishment. (RSA 2048 or 3072-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 (DHE 2048, ECDHE P-256, P-384, P-521)
5	TLS HMAC Keys	HMAC	TLS integrity and authentication session keys (HMAC-SHA-1, HMAC-SHA2-256, HMAC-SHA2-384, HMAC-SHA2-512)
6	TLS Encryption Keys	AES	TLS encryption session keys (128 and 256 CBC or GCM)
7	SSH Session Integrity Keys	HMAC	Used in all SSH connections to the security module's command line interface. (HMAC-SHA-1, HMAC-SHA2-256, HMAC-SHA2-512)

8	SSH Session Encryption Keys	AES	Used in all SSH connections to the security module’s command line interface. (128, 192, and 256 bits in CBC and CTR, or 128 and 256 bits in GCM)
9	SSH DH Private Components	DH	Diffie Hellman private component used in key establishment (DH 2048)
16	Firmware code integrity check	HMAC ECDSA	Used to check the integrity of crypto-related code. (HMAC-SHA-256 and ECDSA P-256) *Keys used to perform power-up self-tests are not CSPs and do not need to be zeroized as per IG 7.4
17	Firmware Content Encryption Key	AES-256	Used to encrypt/decrypt firmware, software, and sensitive content.
18	Password	Password	Authentication string with a minimum length of 6 characters. Stored hashed with SHA-256 and nonce.
19	DRBG Seed /State	DRBG	AES 256 CTR DRBG used in the generation of a random values.

The TOE performs a key error detection check on each internal, intermediate transfer of a key. The TOE stores persistent secret and private keys in encrypted form (AES encrypted) when not in use. The KEK (Key Encryption Key) is the Firmware Content Encryption Key (also known as the Master Key). The KEK is not stored encrypted but is protected either using 1.) Cryptod (Palo Alto Networks proprietary keys storage module) 2.) External HSM. By default, HSM (hardware security module) is not used as it needs to be configured. If stored by Cryptod (evaluated configuration), then it is destroyed by the TOE’s overwriting method. If it is stored via External HSM (operational environment), it is protected by the HSM and is out of scope. 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, KEK, hashed passwords, and entropy seeds. Note: Only the KEK is stored in plaintext and is zeroized as noted above. It is used to encrypt all the private keys and other sensitive data.

	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, followed by a read-verify. Sensitive data in volatile memory includes session keys such as encryption keys, integrity keys, pre-Master secret, etc.
FCS_HTTPS_EXT.1/ * FCS_TLSC_EXT.1 FCS_TLSC_EXT.2 FCS_TLSS_EXT.1 FCS_TLSS_EXT.2	The TOE can be configured as a TLS server for mutual certificate-based authentication for secure connections. To enable certificate-based authentication, the TOE must be configured to use a client certificate profile using the Device > Certificate Management > Certificate Profile tab. The TOE uses TLS service profiles to specify a certificate and the allowed protocol versions for TLS services. The TOE (as a TLS client) uses TLSv1.2 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, firewall, Wildfire). The key agreement parameters of the server key exchange message consist of the key establishment parameters generated by the TOE: RSA with key size of 2048 bits and 3072 bits, Diffie-Hellman parameters with key size of 2048 bits, ECDSA implementing NIST curves secp256r1 and secp384r1. The TOE denies connections from clients requesting connections using SSL 2.0, SSL 3.0, or TLS 1.0 and shall not establish a trusted channel if the distinguished name (DN) or Subject Alternative Name (SAN) contained in a certificate does not match the expected identifier for the peer. The TOE can be configured as a TLS server to permit inbound remote administration traffic (HTTPS) in which the peer initiates handshake and peer authentication is performed via username and password credentials. The TOE's HTTPS protocol complies with RFC 2818 and is implemented using TLS 1.2 (RFC 5246) and TLS 1.1 (RFC 4346). The key agreement parameters of the server key exchange message consist of the key establishment parameters generated by the TOE: RSA with key size of 2048 bits and 3072 bits, Diffie-Hellman parameters with key size 2048 bits, ECDSA implementing NIST curves secp256r1 and secp384r1. The TOE denies connections from clients requesting connections using SSL 2.0, SSL 3.0, or TLS 1.0. The TOE can be configured as a TLS client for secure communication to an external audit server. The TOE presents the Supported Elliptic Curves Extension in the Client Hello with the secp256r1, secp384r1, and secp521r1 NIST curves and is configured when FIPS-CC mode is enabled. The TOE verifies that the presented identifier matches the reference identifier according to RFC 6125 and only establishes a trusted channel if the peer certificate is valid. The TOE compares the external server's presented identifier to the reference identifier by matching the certificate FQDN (hostname) or IPv4/IPv6 address in the SAN field or CN (of subject Field) of the server certificate. The SAN is checked first and if there is any match, the connection is allowed. The TOE supports wildcards for peer authentication using FQDN (hostname) only. The only supported IP address format for IPv4 is specified in RFC 3986 and IPv6 is specified in RFC 5952. Certificate pinning is not supported but mutual authentication is supported. The TOE implements TLS 1.2 (RFC 5246) and TLS 1.1 (RFC 4346). TOE (as TLS client) to syslog server (same ciphersuites for mutual authentication if configured). Support TLSv1.2 only and RSA, DHE (finite-field based), and ECDHE (elliptic curve-based) schemes. • TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268 • TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268 • TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268 • TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268 • TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492

	• TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492 • TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492 • TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492 • TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246 • TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246 • TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246 • TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246 • TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288 • TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288 • 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 TOE (as TLS server) for the Web UI management connection (same for mutual authentication). Supports TLSv1.1 or TLSv1.2 only, and DHE (finite-field based) and ECDHE (elliptic curve-based) schemes. • TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268 • TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268 • TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492 • TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492 • TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 • TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 TOE (as TLS server) connection to firewall or Wildfire (mutual authentication required). Supports TLSv1.1 or TLSv1.2 only, and DHE (finite-field based), and ECDHE (elliptic curve-based) schemes. • TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268 • TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268 • TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492 • TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492 • TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492

	• TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492 • TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256 as defined in RFC 5246 • TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246 • 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
FCS_SSHS_EXT.1	The TOE supports SSHv2 (compliant to RFCs 4251, 4252, 4253, 4254, 4344, 5656, and 6668) with AES encryption/decryption algorithms (aes128cbc, aes256-cbc, aes128-ctr, aes256-ctr, aes128-gcm@openssh.com, aes256-gcm@openssh.com) with key sizes of 128 and 256 bits. No optional characteristics are supported. The TOE also supports HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-512, aes128-gcm@openssh.com, and aes256-gcm@openssh.com for integrity and authenticity. Both encryption and integrity algorithms are administrator-configurable and while 3DES, HMAC-MD5, diffie-hellman-group-1 are also supported, they are all disabled when FIPS-CC mode is enabled. Only the Approved encryption and integrity algorithms along with key exchange algorithms diffie-hellman-group14-sha1, ecdh-sha2-nistp256, ecdh-sha2-nistp384, and ecdh-sha2-nistp521 and authentication public-key algorithm ssh-rsa are permitted in the evaluated configuration. If the SSH client (in the operational environment) only support non-Approved algorithms, the SSH connection will be rejected by the TOE. The TOE uses Palo Alto Networks Crypto Module implementation to support the SSHv2 connections. The authentication timeout period is 60 seconds allowing clients to retry only 4 times. In addition, both public-key (RSA) and password-based authentication can be configured with password-based being the default method. The SSH packets are limited to 256 Kbytes 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 configurable amount of data (10 – 4000 MBs) or time (10 – 3600 seconds) has passed, whichever threshold occurs first. In the evaluated configuration, the administrator should not configure the SSH data rekey threshold to be more than 1024 MBs.

6.3 Identification and Authentication

FIA_UIA_EXT.1 FIA_UAU_EXT.2 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 user name (i.e., user identity or ID), a password or X.509v3 certificate/common access card, and a role. To enable client certificate-based authentication (i.e., mutual authentication), the TOE must be configured to use a client certificate profile using the Panorama > Certificate Management > Certificate Profile tab. When a client certificate profile is enabled, each administrator must use a client certificate for access to the TOE via TLS. The client certificate must identify the domain name (in this case, the username) in the SAN (first) or CN (second, if SAN is not present). The TOE will match the presented username to the username in the local database and associated role. Only one role is specified in the user account per user. The TOE uses the user name and password attributes to identify and authenticate the user when the user logs in via the GUI or CLI. 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 CLI or UI, the default authentication method is password. The administrators must configure public-key authentication which is supported for both SSH and HTTPS sessions. It uses the role attribute to specify user permissions and control what the user can do with the GUI or CLI. The administrator can logon to the GUI by using a secure connection (HTTPS) from a web browser or to CLI by using a secure connection (SSHv2) from a SSH client. The TOE provides access to the GUI/CLI locally via direct RJ-45 Ethernet cable connection and remotely using an HTTPS/TLS or SSHv2 client. The administrator enters the IP address of the TOE and their username and password. The TOE also can be configured to require a client certificate (mutual authentication) and additionally require the username and password or not (i.e., 2-factors authentication). The credentials may be supplied by a CAC or retrieved from the client computer. 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.
FIA_PMG_EXT.1	Passwords can be composed of upper and lower case letters, numbers and special characters (“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(”, “)”, “'”, “+”, “,”, “-”, “.”, “/”, “:”, “;”, “<”, “=”, “>”, “[”, “\”, “]”, “_”, “`”, “{”, “}”, and “~”). The minimum password length is configurable by the administrator up to a maximum length of 31 characters. Note in FIPS-CC mode, the minimum and maximum length is from 6 (up to 15) to 31 characters.
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) of unsuccessful authentication attempts. The lock can be configured to last a specified amount of time (1 – 60 minutes) during which providing the correct credentials will still not allow access (i.e., locked out), or until an administrator log in to unlock the locked user. These settings can be configured for both HTTPS/TLS and SSH remote administration connections but applies to password authentication only. Public-key authentication is not vulnerable to weak passwords that can be brute-forced. It’s recommended that at least one administrator, preferably the Superuser role (predefined ‘admin’ account), is configured with public-key authentication for SSH.

	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 and unlock users. In addition, the user can also wait until the lockout time expires.
FIA_X509_EXT.1/Rev FIA_X509_EXT.2/*	The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for TLS (server authentication and mutual authentication) and HTTPS connections. Public key infrastructure (PKI) credentials, such as Rivest, Shamir, and Adelman (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 X509 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. The TOE supports Online Certificate Status Protocol (OCSP) and Certificate Revocation List (CRL) status verification for certificate profiles. If both are configured, the devices first try the OCSP method; if the OCSP server is unavailable, the devices use the CRL method. The TOE uses the following rules for validating the extendedKeyUsage[2] field: • Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field. • Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field. • OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field. 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 Administrator must import or generate a root CA certificate and store it in the TOE trust store. To use only a specific trusted certificate, the Administrator must specify only that certificate in the Certificate Profile and tied that Profile to a TLS connection. 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, you must configure the TOE to access an OCSP responder (server). The entity that

[2] Certificates are not used for trusted updates or executable code integrity.

	manages the OCSP responder can be a third-party certificate authority (CA) or, if your enterprise has its own PKI, the TOE itself. When establishing a TLS session, clients can use OCSP to check the revocation status of the authentication certificate. The authenticating client sends a request containing the serial number of the certificate to the OCSP responder (server). The responder searches the database of the certificate authority (CA) that issued the certificate and returns a response containing the status (good, revoked or unknown) to the client. The advantage of the OCSP method is that it can verify status in real-time, instead of depending on the issue frequency (hourly, daily, or weekly) of CRLs. The TOE downloads and caches the last-issued CRL for every CA listed in the trusted CA list of the TOE. Caching only applies to validated certificates; if a TOE never validated a certificate, the TOE cache does not store the CRL for the issuing CA. Also, the cache only stores a CRL until it expires. The TOE supports CRLs only in Distinguished Encoding Rules (DER) or PEM format. When the certificate status is unknown or cannot be determined, the TLS session is blocked. This is the default for syslog connection and cannot be changed. For the TLS management session (if mutual authentication is configured) and for the TLS sessions to the firewall and Wildfire, the behavior is to block the TLS session.
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 check to make sure the certificate chain are present in the TOE. Otherwise, the TOE will reject the certificate and will not associate it with the CSR.

6.4 Security Management

FMT_MOF.1/ManualUpdate FMT_MTD.1/CoreData	The TOE provides a GUI management interface and CLI to support security management of the TOE. The GUI is accessible via direct connection to the management port on the device (local access), or remotely over HTTPS. Note the TOE in Log Collector mode does not support GUI. The CLI is accessible via direct connection to the management port on the device (local access), or remotely over SSHv2. The restricted role-based privileges enable only authorized administrators to configure the TOE functions such as updating the TOE and manipulating TSF data. For example, the ability to manage the TOE's trust store is restricted to Administrators only. 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).
FMT_SMF.1	The security management functions provided by the TOE include, but are not limited to: • Ability to administer the TOE locally and 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 set the time which is used for time-stamps; • 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 The GUI, CLI, and API (XML and REST) provide the same supported management functionality. With regards to the TSF management functions above, they are available on both interfaces. The local interface supports the use of a dedicated Ethernet port that only supports communication with a whitelisted local IP address.
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 and in Panorama and Management-Only modes, the TOE has the following pre-defined administrator roles: Superuser, Superuser (Read-Only), and Panorama Administrator. These administrator roles (except Read-Only) are all considered Security Administrator as defined in the [NDcPP] for the purposes of this ST. For example, a user with Superuser role can create, modify, or delete user accounts but user with Read-Only role cannot. All roles can administer the TOE both locally and remotely. In Log

	Collector mode, only the Superuser role is supported and there is only one user account. • **Superuser**—Full read-write access to Panorama and all device groups, templates, and managed firewalls including user and role management (create, modify, delete). • **Superuser (Read-Only)**—Read-only access to Panorama and all device groups, templates, and managed firewalls. • **Panorama Administrator**—Full access to Panorama except for the create, modify, or delete administrators or roles.

6.5 Protection of the TSF

FPT_SKP_EXT.1 FPT_APW_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 X509 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_TST_EXT.1	The TOE meets self-test requirements and therefore provides self-tests at start-up to demonstrate the correct operation of: key error detection, cryptographic algorithms, and RNG. Conditional self-tests are also run during the course of normal operation. The self-tests verify the integrity of stored TSF executable code and TSF data. The TOE performs the following Power-on self-tests: • AES Encrypt Known Answer Test • AES Decrypt Known Answer Test • AES GCM Encrypt Known Answer Test • AES GCM Decrypt Known Answer Test • AES CCM Encrypt Known Answer Test • AES CCM Decrypt Known Answer Test • RSA Sign Known Answer Test • RSA Verify Known Answer Test • RSA Encrypt/Decrypt 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 • SHA-256 Known Answer Test • SHA-384 Known Answer Test • SHA-512 Known Answer Test • DRBG SP800-90A Known Answer Tests • SP 800-90A Section 11.3 Health Tests • DH Known Answer Test

	- ECDH Known Answer Test - Firmware Integrity Test – verified with HMAC-SHA-256 and ECDSA P-256. If the calculated result does not equal the previously generated result, the software/firmware test shall fail. 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 TOE performs the following Conditional Self-Tests within the cryptographic module when the conditions specified for the tests occur: 1. Continuous Random Number Generator (RNG) test – performed on NDRNG and DRBG 2. RSA Pairwise Consistency Test 3. ECDSA Pairwise Consistency Test 4. Firmware Load Test – Verify using RSA 2048 with SHA-256 signature on firmware at time of load. If the digital signature cannot be verified, the test shall fail. The RNG continuous random number generator test is performed on each RNG and tests for failure to a constant value as follows: 1. If each call to a RNG produces blocks of n bits (where n > 15), the first n-bit block generated after power-up, initialization, or reset shall not be used, but shall be saved for comparison with the next n-bit block to be generated. Each subsequent generation of an n-bit block shall be compared with the previously generated block. The test shall fail if any two compared n-bit blocks are equal. 2. If each call to a RNG produces fewer than 16 bits, the first n bits generated after power-up, initialization, or reset (for some n > 15) shall not be used, but shall be saved for comparison with the next n generated bits. Each subsequent generation of n bits shall be compared with the previously generated n bits. The test fails if any two compared n-bit sequences are equal. The TOE performs the following pair-wise consistency tests for public and private keys: 1. If the keys are used to perform an approved key transport method or encryption, then the public key shall encrypt a plaintext value. The resulting ciphertext value shall be compared to the original plaintext value. If the two values are equal, then the test shall fail. If the two values differ, then the private key shall be used to decrypt the ciphertext and the resulting value shall be compared to the original plaintext value. If the two values are not equal, the test shall fail. 2. If the keys are used to perform the calculation and verification of digital signatures, then the consistency of the keys shall be tested by the calculation and verification of a digital signature. If the digital signature cannot be verified, the test shall fail. If a self-test fails, the TOE enters an error state and outputs an error indicator. The TOE doesn’t perform any cryptographic operations 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 a maintenance state in which the reason for the reboot can be determined.
FPT_TUD_EXT.1	Authorized administrators may query the current software/firmware version of the TOE (command ‘show system info | match sw-version’). 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-2048) and encrypt (using AES-256) 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 update server 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 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. The clock is used for audit record time stamps, measuring session activity for termination, and for cryptographic operations based on time/date.

6.6 TOE Access

FTA_SSL_EXT.1	The TOE subsequently will enforce an administrator-defined inactivity timeout value after which the inactive, local or remote, session will be terminated regardless of authentication methods (e.g., password, public-key, x509v3 certificate). The TOE can be configured by an administrator to set an interactive session timeout value (any integer value from 1 to 60 minutes). The function is disabled 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 issuing from the remote client) for the defined timeout value will be terminated. A local session that is similarly inactive for the defined timeout period 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.
FTA_SSL.3	
FTA_SSL.4	The TOE provides both local and remote users the ability to logout (or terminate) their sessions as directed by the user.
FTA_TAB.1	The TOE can be configured to display an informative banner that will appear prior to authentication when accessing the TOE via either a direct or remote connection to the management port in order to access the Web Interface (HTTPS) or CLI (SSH).

6.7 Trusted Path/Channels

FTP_ITC.1	The TOE can be configured to send audit records to external Syslog server(s) using TLS in real-time. The TOE permits the TSF to initiate communication with the Syslog server, firewall, and Wildfire using TLS trusted channel. The TOE communicates with its authorized entities over TLS only and all communication are sent over the trusted channel, including the TOE initial communication. The underlying TLS algorithms are supported by CAVP-validated cryptographic mechanisms included in the TOE implementation.
FTP_TRP.1/Admin	The TOE provides SSH and HTTPS (TLSv1.1 and TLSv1.2) to support secure remote administration. HTTPS is supported in Panorama and Management-Only modes. 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 this secure channel. In FIPS-CC mode, telnet and HTTP are disabled permanently.

7. Protection Profile Claims

This ST is conformant to the [NDcPP].

8. Rationale

This security target includes by reference the [NDcPP] Security Problem Definition, Security Objectives, and Security Assurance Requirements. The security target makes no additions to the [NDcPP] assumptions. Security functional requirements have been reproduced verbatim with the protection profile operations completed. Operations on the security requirements follow [NDcPP] application notes and assurance activities. The security target did not add or remove any security requirements. Consequently, [NDcPP] rationale applies and is complete.