Palo Alto Networks PA-400 Series, PA-800 Series, PA-1400 Series, PA-3200 Series, PA-3400 Series, PA-5200 Series, PA-5400 Series, PA-5450, PA-7000 Series, and VM Series Next-Generation Firewall with PAN-OS 11.2 Security Target

Version: 1.0
Date: July 8, 2024

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 the next-generation firewall running PAN-OS v11.2 provided by Palo Alto Networks Inc.

The next-generation firewall includes the PA-410, PA-410R-5G, PA-415, PA-415-5G, PA-440, PA-445, PA-450, PA-450R, PA-450R-5G, PA-455, PA-455-5G, PA-460, PA-820, PA-850, PA-1410, PA-1420, PA-3220, PA-3250, PA-3260, PA-3410, PA-3420, PA-3430, PA-3440, PA-5220, PA-5250, PA-5260, PA-5280, PA-5410, PA-5420, PA-5430, PA-5440, PA-5445, PA-5450, PA-7050, PA-7080, and PA-7500 appliances and the virtual appliances in the VM-Series VM-50, VM-100, VM-300, VM-500, VM-700 which are used to manage enterprise network traffic flows using function specific processing for networking, security, and management. The next-generation firewalls identify which applications are flowing across the network, irrespective of port, protocol, or location.

The focus of this evaluation is on the TOE functionality supporting the claims in the collaborative Protection Profile for Network Devices [NDcPP], PP-Module for Stateful Traffic Filter Firewalls [FW-Module], PP-Module for Intrusion Prevention System [IPS-Module], Functional Package for Secure Shell [SSHPKG], and PP-Module for Virtual Private Network (VPN) Gateways [VPNGW-Module] as amended by CSfC Selections for VPN Gateways [CSfC]. The CSfC Selections for VPN Gateways are specified in the following NSA's website: https://www.nsa.gov/Portals/70/documents/resources/everyone/csfc/components-list/selections/vpn-gateways.pdf

The only capabilities covered by the evaluation are those specified in the aforementioned Protection Profiles, all other capabilities are not covered in the evaluation. The security functionality specified in [NDcPP], [FW-Module], [IPS-Module], [SSHPKG], and [VPNGW-Module] includes protection of communications between TOE components and trusted IT entities, identification and authentication of administrators, auditing of security-relevant events, ability to verify the source and integrity of updates to the TOE, the implementation of firewall-related security features, the termination of IPsec VPN tunnels, and specifies CAVP-validated cryptographic mechanisms.

The Security Target contains the following additional sections:

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

1.1 Security Target, TOE and CC Identification

ST Title – Palo Alto Networks PA-400 Series, PA-800 Series, PA-1400 Series, PA-3200 Series, PA-3400 Series, PA-5200 Series, PA-5400 Series, PA-5450, PA-7000 Series, and VM Series Next-Generation Firewall with PAN-OS 11.2 Security Target

ST Version – Version 1.0

ST Date – July 8, 2024

TOE Identification – Palo Alto Networks PA-400 Series, PA-800 Series, PA-1400 Series, PA-3200 Series, PA-3400 Series, PA-5200 Series, PA-5400 Series, PA-5450, PA-7000 Series, and VM Series, Next-Generation Firewall with PAN-OS v11.2. The specific Firewall appliance models include:

1. PA-400 Series
    a. PA-410
    b. PA-410R-5G
    c. PA-415
    d. PA-415-5G
    e. PA-440
    f. PA-445
    g. PA-450
    h. PA-450R
    i. PA-450R-5G
    j. PA-455
    k. PA-455-5G
    l. PA-460
2. PA-800 Series
    a. PA-820
    b. PA-850
3. PA-1400 Series
    a. PA-1410
    b. PA-1420
4. PA-3200 Series
    a. PA-3220
    b. PA-3250
    c. PA-3260
5. PA-3400 Series
    a. PA-3410
    b. PA-3420
    c. PA-3430
    d. PA-3440
6. PA-5200 Series
    a. PA-5220
    b. PA-5250
    c. PA-5260
    d. PA-5280[1]
7. PA-5400 Series
    a. PA-5410

[1] PA-5280 can operate in Express or Secure mode. Secure mode just means it's 5G-ready and requires a license upgrade.

   b. PA-5420
   c. PA-5430
   d. PA-5440
   e. PA-5445
8. PA-5450[2]
9. PA-7000 Series[3]
   a. PA-7050
   b. PA-7080
   c. PA-7500
10. VM-Series
   a. VM-50
   b. VM-100
   c. VM-300
   d. VM-500
   e. VM-700

The Palo Alto VM-Series is supported on the following hypervisors:

- VMware
  - VMware ESXi with vSphere 7.0
- Linux KVM
  - Ubuntu: 20.04 LTS
- Microsoft Hyper-V Server 2019 ---- The VM-Series firewall can be deployed on a server running Microsoft Hyper-V. Hyper-V is packaged as a standalone hypervisor, called Hyper-V Server 2019, or as an add-on/role for Windows Server 2019.

Evaluation testing included the following:

VMware ESXi 7.0:

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

Microsoft Hyper-V Server 2019:

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

Linux KVM 4 Ubuntu 20.04:

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

Evaluation testing included the following hardware and processors:

[2] PA-5450 firewall supports the following cards: PA-5400 MPC-A, PA-5400 NC-A, and PA-5400 DPC-A.
[3] Palo Alto Networks PA-7000 Series firewalls support different Network Processing Cards (NPC) and Switch Management Cards (SMC): PAN-PA-7050-SMC-B, PAN-PA-7080-SMC-B, PAN-PA-7000-LFC-A, PAN-PA-7000-100G-NPC-A-K2-EXP, PAN-PA-7000-100G-NPC-A-K2-SEC, and PAN-PA-7000-100G-NPC.

- PA-3260: Cavium Octeon CN7360 MIPS64 (DP) / Intel Pentium D1517 (MP)
- PA-5450: Intel Xeon D-2187NT (DP/MP)
- PA-5430: AMD EPYC 7642 (DP/MP)

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:

PP-Configuration for Network Devices, Stateful Traffic Filter Firewalls, Virtual Private Network (VPN) Gateways, and Intrusion Prevention System, Version 2.0, 25 April 2024 [CFG_NDcPP-IPS-FW-VPNGW_v2.0] consisting of the following components:

- collaborative Protection Profile for Network Devices, Version 3.0e, 6 December 2023 [NDcPP]
- PP-Module for Stateful Traffic Filter Firewalls, Version 1.4 + Errata 20200625, 25 June 2020 [FW-Module]
- PP-Module for Virtual Private Network (VPN) Gateways, Version 1.3, 16 August 2023 [VPNGW-Module]
- PP-Module for Intrusion Prevention System (IPS), Version 1.0, 11 May 2021 [IPS-Module]
- Functional Package for Secure Shell (SSH) Version 1.0, 13 May2021 [SSHPKG]

The following NIAP Technical Decisions apply to [NDcPP], [FW-Module], [VPNGW-Module], and/or [IPS-Module] and have been accounted for in the ST development and the conduct of the evaluation:

- TD0545: NIT Technical Decision for Conflicting FW rules cannot be configured (extension of Rfl#201837)
  - o The TD updates test evaluation activities that apply to the TOE. [FW-Module]
- TD0551: NIT Technical Decision for Incomplete Mappings of OEs in FW Module v1.4+Errata
  - o This TD is applicable to the TOE but does not affect the ST or evaluation activities. The TD updates Security Problem Definition mappings and rationale in the [FW-Module]
- TD0595: Administrative Corrections to IPS PP-Module
  - o This TD is applicable to the TOE. [IPS-Module]
- TD0682: Addressing Ambiguity in FCS_SSHS_EXT.1 Tests
  - o TD updates test evaluation activities that apply to the TOE. [SSHPKG]
- TD0695: Choice of 128 or 256 bit size in AES-CTR in SSH Functional Package
  - o This TD is not applicable to the TOE because it supports both 128 and 256 bits for AES-CTR. [SSHPKG]
- TD0722: IPS_SBD_EXT.1.1 EA Correction
  - o This TD is applicable to the TOE. [IPS-Module]
- TD0732: FCS_SSHS_EXT.1.3 Test 2 Update
  - o TD updates test evaluation activities that apply to the TOE. [SSHPKG]
- TD0777: Clarification to Selections for Auditable Events for FCS_SSH_EXT.1
  - o This TD is not applicable to the TOE because it logs failure to establish SSH connection. [SSHPKG]
- TD0781: Correction to FIA_PSK_EXT.3 for MOD_VPNGW_v1.3

o TD updates TSS and AGD evaluation activities that do NOT apply to the TOE. [VPNGW-Module]

- TD0811: Correction to Referenced SFR in FIA_PSK_EXT.3 Test
  o TD updates test evaluation activities that do NOT apply to the TOE. [VPNGW-Module]
- TD0824: Aligning MOD_VPNGW 1.3 with NDcPP 3.0E
  o TD updates SFRs that do apply to the TOE. [VPNGW-Module]
- TD0827: Aligning MOD_CPP_FW_v1.4E with NDcPP 3.0E
  o TD updates PP-Module to add support for NDcPPv3.0e. [FW-Module]
- TD0828: Aligning MOD_IPS_v1.0 with NDcPP 3.0E
  o TD updates PP-Module to add support for NDcPPv3.0e. [IPS-Module]
- TD0836: NIT Technical Decision: Redundant Requirements in FPT_TST_EXT.1
  o TD updates SFRs that do apply to the TOE [NDcPP]
- TD0838: NIT Technical Decision: PPK Configurability in FIA_PSK_EXT.1.1
  o TD updates guidance and tests that do apply to the TOE [VPNGW-Module]

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], [FW-Module], [IPS-Module], [SSHPKG], and [VPNGW-Module].
- 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] [FW-Module], [IPS-Module], [SSHPKG], and [VPNGW-Module].
- Selection/Assignment operations completed by the ST author were bolded to show that it was completed by the ST author and not taken as-is from the PP.
- Iteration operations completed by the ST author are identified with (1), (2), and (next number) with descriptive text following the name (e.g. FCS_HTTPS_EXT.1(1) HTTPS Protocol (TLS Server)).
- Refinement operations completed by the ST author are identified in BOLD text.

1.3.1 Terminology

The following terms and abbreviations are used in this ST:

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

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 Policy	Provides the firewall rule sets that specify whether to block or allow network connections.
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.
Virtual System	Virtual systems are separate, logical firewall instances within a single physical Palo Alto Networks firewall. Virtual systems allow the TOE administrator to customize administration, networking, and security policies for network traffic belonging to specific user groupings (such as departments or customers).

1.3.2 Acronyms

APT	Advanced Persistent Threat
AES	Advanced Encryption Standard
API	Application Programming Interface
C2	Command and Control
CBC	Cipher-Block Chaining
CC	Common Criteria
CEM	Common Evaluation Methodology
CM	Configuration Management
CLI	Command Line Interface
CSfC	Commercial Solutions for Classified
DH	Diffie-Hellman
DMZ	Demilitarized Zone
DoS	Denial of Service
DRBG	Deterministic Random Bit Generator
DP	Data Plane
EEPROM	Electrically Erasable Programmable Read-Only Memory
FIPS	Federal Information Processing Standard
FSP	Functional Specification
FTP	File Transfer Protocol
FW	Firewall
GCM	Galois/Counter Mode
GUI	Graphical User Interface
HMAC	Hashed Message Authentication Code
HTTPS	Hypertext Transfer Protocol Secure
ICMP	Internet Control Message Protocol
IKE	Internet Key Exchange
IP	Internet Protocol
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
IPsec	Internet Protocol Security
MP	Management Plane
NAT	Network Address Translation
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
UDP	User Datagram Protocol
URL	Uniform Resource Locator
VLAN	Virtual Local Area Network
VM	Virtual Machine
VPN	Virtual Private Network
VPNGW	Virtual Private Network Gateway

2. Product Description

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 products below are considered trusted IT products in the operational environment and only the secure communication (FTP_ITC.1) between the firewalls and the products are claimed and validated in this evaluation. The product descriptions below are provided for completeness only.

- GlobalProtect safeguards the mobile workforce by inspecting all traffic using the organization's next-generation firewalls that are deployed as internet gateways, whether at the perimeter, in the DMZ, or in the cloud. Laptops, smartphones and tablets with the GlobalProtect app automatically establish a secure TLS/IPsec VPN connection to the next-generation firewall with the best performance for a given location, thus providing the organization with full visibility of all network traffic, for applications, and across all ports and protocols.

2.1 TOE Overview

The Target of Evaluation (TOE) is comprised of one instance of the Palo Alto Networks next-generation firewall that includes the Palo Alto Networks PA-410, PA-410R-5G, PA-415, PA-415-5G, PA-440, PA-445, PA-450, PA-450R, PA-450R-5G, PA-455, PA-455-5G, PA-460, PA-820, PA-850, PA-1410, PA-1420, PA-3220, PA-3250, PA-3260, PA-3410, PA-3420, PA-3430, PA-3440, PA-5220, PA-5250, PA-5260, PA-5280, PA-5410, PA-5420, PA-5430, PA-5440, PA-5445, PA-5450, PA-7050, PA-7080, and PA-7500 appliances and the virtual appliances in the VM-Series VM-50, VM-100, VM-300, VM-500, VM-700 with PAN-OS v11.2. The next-generation firewall provides policy-based application visibility and control to protect traffic flowing through the enterprise network.

The 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 connections to other next-generation firewalls or third-party security devices.

A next-generation firewall is typically installed between an edge router or other device facing the Internet and a switch or router connecting to the internal network. The Ethernet interfaces on the firewall can be configured to support various networking environments, including: Layer 2 switching and VLAN environments; Layer 3 routing environments; transparent in-line deployments; and combinations of the three. The scope of the evaluation does not cover Layer 2 switching, VLAN, and transparent in-line deployments. In addition, the TOE is not a distributed TOE as defined in the [NDcPP].

The next-generation firewalls provide granular control over the traffic allowed to access the protected network. They allow an administrator to define security policies for specific applications, rather than rely on a single policy for connections to a given port number. For each identified application, the administrator can specify a security policy to block or allow traffic based on the source and destination zones, source and destination addresses, or application services. The next-generation firewalls also support the following types of policy:

- Application-based policies (e.g., FTP)
- User Identification Agent (UIA) – the TOE uses user-specific information provided by UIA in the operational environment for security policy enforcement. The UIA automatically collects user-specific information and provides mapping information between IP addresses and network users, and provides these information to the TOE which then uses mappings in its security policy enforcement. The user ID can be an attribute specified in the TOE security policies upon which they are enforced. The UIA works with both Ipv4 addresses and Ipv6 addresses.

Security policies can include specification of one or more security profiles, which provide additional protection and control. Security profiles are configured and applied to firewall policy. Each security policy can specify one or more of the following security profiles:

- Vulnerability Protection profiles
- Zone Protection profiles
- DoS Protection profiles
- IKE Crypto Security profiles
- IPsec Crypto Security profiles

The next-generation firewall products provide the following features:

- Application-based policy enforcement — the product uses a traffic classification technology named App-ID to classify traffic by application content irrespective of port or protocol. Protocol and port can be used in conjunction with application identification to control what ports an application is

allowed to run on. High risk applications can be blocked, as well as high-risk behavior such as file-sharing or FTP.

- Threat prevention — the firewall includes threat prevention capabilities (i.e., Vulnerability Protection profile, Zone Protection profile) that can protect the network from viruses, worms, spyware, and other malicious traffic.
- DoS Protection – the firewall is designed to protect against flooding attack within the protected network. The DoS Protection profile also specifies the maximum connection per second (CPS) rate and how long a blocked IP address remains on the Block IP list.
- IKE Crypto Security profiles – specify protocols and algorithms for identification, authentication, and encryption (IKEv2, Phase 1).
- IPsec Crypto Security profiles – specify protocols and algorithms for authentication and encryption in VPN tunnels based on IPSec SA negotiation (Phase 2).
- Threat intrusion prevention feature protects and defends the network from commodity threats and advanced persistent threats (APTs) using multi-pronged detection mechanisms to combat the entire gamut of the threat landscape. The core Threat Prevention license subscription is based on signatures generated from malicious traffic data collected from various Palo Alto Networks services. These signatures are used by the firewall to enforce security policies based on specific threats, which include command-and-control (C2), various types of known malware, and vulnerability exploits. As a part of the threat mitigation policies, administrator can also identify and block known or risky file types and IP addresses, of which several premade categories are available, including lists specifying bulletproof service providers and known malicious IPs. In cases where specialized tools and software are used, administrators can create their own vulnerability signatures to customize the intrusion prevention capabilities to their network's unique requirements.
- Management — each firewall can be managed through a Graphical User Interface (GUI), API, or CLI. The interface provides an administrator with the ability to establish policy controls, provide the means to control what applications network users are allowed access to, and to control logging. When configured in a FIPS-CC mode of operation, the GUI and API are secured using HTTP over TLS (HTTPS) and CLI is secured using SSH.

Firewall Policy Enforcement

The App-ID classification technology uses four classification techniques to determine exactly what applications are traversing the network irrespective of port number. As traffic flows through the TOE, App-ID identifies traffic using the following classification engines.

- Application Protocol/Port: App-ID identifies the protocol (such as FTP) and the port number of the traffic. Protocol/Port information is primarily used for policy enforcement, such as allowing or blocking a specific application over a specific protocol or port number, but is sometimes used in classification, such as ICMP traffic where the protocol is the primary classification method used.
- Application Protocol Decoding: App-ID's protocol decoders determine if the application is using a protocol as a normal application transport (such as HTTP for web browsing applications), or if it is only using the apparent protocol to hide the real application protocol (for example, Yahoo! Instant Messenger might hide inside HTTP).

Threat Prevention

The next-generation firewall includes a real-time threat prevention engine that inspects the traffic traversing the network for a wide range of threats. The threat prevention engine scans for all types of threats with a uniform signature format and can identify and block a wide range of threats across a broad set of applications in a single pass. The threats that can be detected by the threat prevention engine include: viruses; spyware (inbound file scanning, and connections to infected web sites); application vulnerability exploits; and phishing/malicious URLs.

App-ID and Threat Prevention Signature Updates

App-ID and threat prevention signatures (collectively known as content updates) may be updated periodically using the dynamic updates feature of the firewall. The TOE can be configured to contact Palo Alto Networks' updates.paloaltonetworks.com to download new content updates as they are made available. The connection to the updates.paloaltonetworks.com is secured with TLS using FIPS-approved algorithms.

Management

The next-generation firewall provides secure connections for the Web/CLI/API Management interface. The Web, API, and command interfaces provide administrators with the ability to manage, configure and monitor the TOE.

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 and CC requirements for self-tests as well as X.509 certificates checks are enforced. In this mode, only CC Approved cryptographic algorithms and key sizes are available.

2.2 TOE Architecture

The firewalls' architecture is divided into two subsystems: the control plane and the data plane. The control plane provides system management functionality while the data plane handles all data processing on the network; both reside on the firewall appliance. The TOE relies on the User Identification Agent installed on a separate dedicated PC in the operational environment to retrieve user-specific information that is used for policy enforcement.

The following diagram depicts both the TOE and the User Identification Agent:

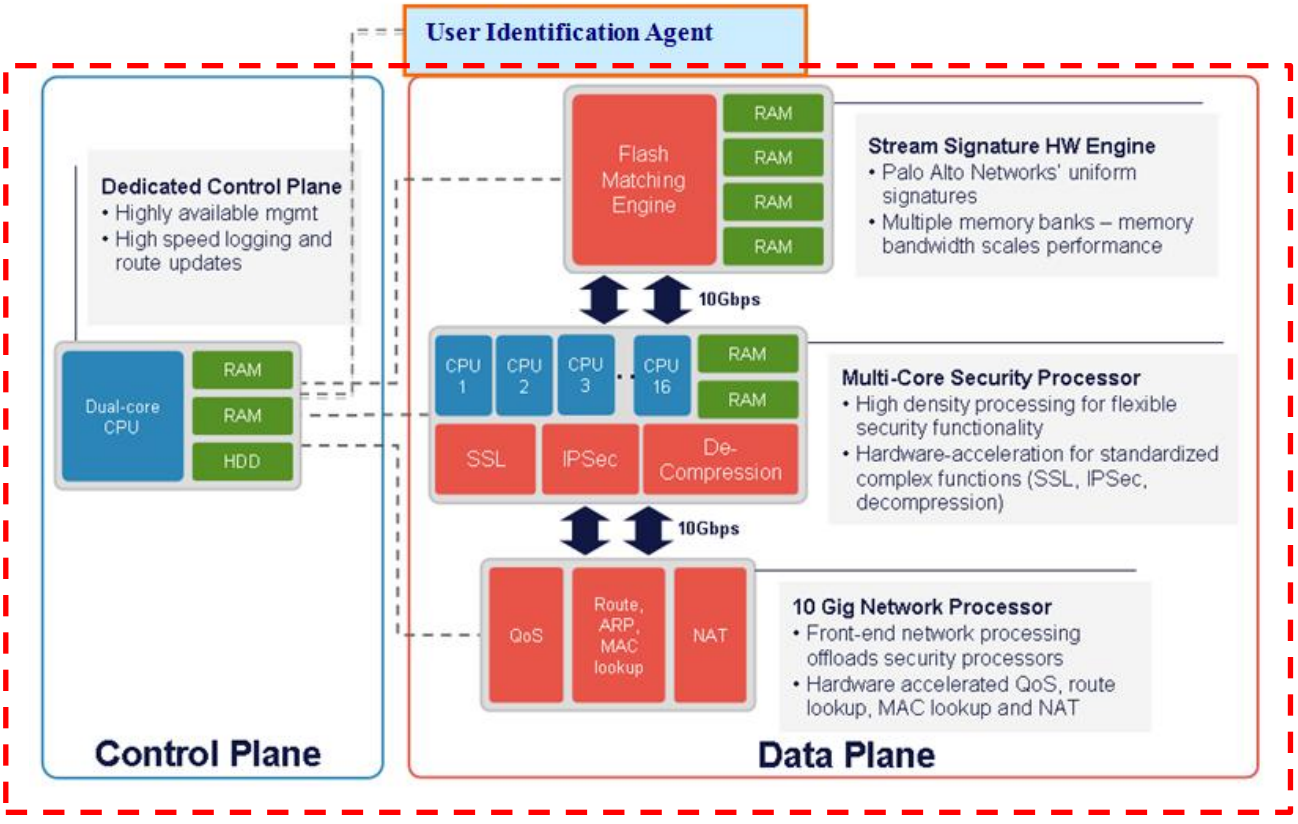

Figure 1: TOE Architecture

The control plane includes a multi-core CPU, with dedicated memory and a hard drive for local log, configuration, and software storage. The data plane includes three components—the network processor, the security processor, and the stream signature processor—each with its own dedicated memory and hardware processing.

In summary, the functionality provided by each component of the system is as follows:

Control Plane (also known as Management Plane)

The control plane provides all device management functionality, including:

- All management interfaces – provide a secure connection for the Web Interface GUI/API and CLI on SSH.
- Configuration management of the device, such as controlling the changes made to the device configuration, as well as the compilation and pushing to the Data Plane of a configuration change.
- Logging infrastructure for traffic, threat, alarm, configuration, and system logs.
- Administration controls, including administrator authentication and audit trail information for administrators logging in, logging out, and configuration changes.
- Interactions with the UIA to retrieve the user to IP address mapping information that is used for policy enforcement (via the Data Plane).

Data Plane (DP)

The data plane provides all data processing and security detection and enforcement, including:

- All networking connectivity, packet forwarding, switching, routing, and network address translation
- Application identification, using the content of the applications, not just port or protocol
- Application decoding, threat scanning for all types of threats and threat prevention
- Policy lookups to determine what security policy to enforce and what actions to take, including logging
- Denial of Service (DoS) protection including TCP Sync flooding attack
- Logging, with all logs sent to the control plane for processing and storage

Site-to-site IPsec VPN supports Ipv4 or Ipv6 site-to-site connections. That is, administrator can establish IKE and IPsec Security Associations (Sas) between Ipv4 or Ipv6 endpoints. The web interface can be used to enable, disable, restart, or refresh an IKE gateway or an IPsec VPN tunnel to simplify troubleshooting.

VM-Series

The VM-Series on specified hardware supports the exact same next-generation firewall and advanced threat prevention features that are available in the physical form factor appliances, allowing an administrator to safely enable applications flowing into, and across private, public and hybrid cloud computing environments.

Each VM-Series virtual appliance in its evaluated configuration is installed on a hardware platform as specified in Section 1.1 that includes a Vmware, Linux KVM, or Microsoft Hyper-V hypervisor and an Intel Core or Xeon processor based on the Skylake, Cascade Lake, Ivy Bridge, Haswell, or Broadwell microarchitectures that implement Intel Secure Key, and Network Interface Controllers supported by the Server.

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.
- Virtualized Firewalls installed on specified hardware – the VM-Series supports the exact same next-generation firewall and advanced threat prevention features available in the physical form factor appliances, allowing an administrator to safely enable applications flowing into, and across their private, public and hybrid cloud 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 PAN-OS VMs. VMs are deployed in the system using Intel CPUs.
- PAN-OS v11.2 – the software/firmware component that runs the appliance. For VMs PAN-OS is software and for hardware appliances PAN-OS is firmware. PAN-OS is built on top of a Linux kernel and runs along with NGINX (the web server that Palo Alto Networks uses), crond, syslogd, and various vendor-developed applications that implement PAN-OS capabilities. PAN-OS provides the logical interfaces for network traffic. PAN-OS runs on both the Control Plane and the Data Plane and provides all firewall functionalities provided by the TOE, including the threat prevention capabilities as well as the identification and authentication of users and the management functions. PAN-OS provides unique functionality on the two planes based on the applications that are executing. The Control Plane provides a GUI Web management interface to access and manage the TOE functions and data. The Data Plane provides the external interface between the TOE and the external network to monitor network traffic so that the TSF can enforce the TSF security policy.

The physical boundary of the TOE comprises the firewall appliance (PA-410, PA-410R-5G, PA-415, PA-415-5G, PA-440, PA-445, PA-450, PA-450R, PA-450R-5G, PA-455, PA-455-5G, PA-460, PA-820, PA-850, PA-1410, PA-1420, PA-3220, PA-3250, PA-3260, PA-3410, PA-3420, PA-3430, PA-3440, PA-5220, PA-5250, PA-5260, PA-5280, PA-5410, PA-5420, PA-5430, PA-5440, PA-5445, PA-5450, PA-7050, PA-7080, and PA-7500); and the virtual appliances on specified hardware in the VM-Series VM-50, VM-100, VM-300, VM-500, VM-700. The next-generation firewall models differ in their performance capability, but they provide the same security functionality.

Virtual systems are supported on PA-400 Series, PA-3200 Series, PA-3400 Series, PA-5200 Series, PA-5400 Series, and PA-7000 Series firewalls. A Virtual Systems license is required to support multiple virtual systems on PA-400 Series, PA-3200 Series, and PA-3400 Series firewalls, and to create more than the base number of virtual systems supported on a platform. The PA-800 cannot support virtual systems. Virtual systems specify a collection of physical and logical firewall interfaces that should be isolated. Each virtual system contains its own security policy and its own set of logs that will be kept separate from all other virtual systems.

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

- PA-410/PA-410R-5G: 7 RJ-45 10/100/100 ports for network traffic (Ethernet ports); 1 RJ-45 port to access the device GUI through an Ethernet interface (management ports); and 1 RJ-45 port for connecting a serial console (management console port); 2 USB (disabled in FIPS-CC mode except for power) and Power input. Processor: Intel Atom C3436L (DP/MP)
- PA-415/PA-415-5G/PA-445: 10 RJ-45 10/100/100 ports for network traffic (Ethernet ports); 1 RJ-45 port to access the device GUI through an Ethernet interface (management ports); and 1 RJ-45 port for connecting a serial console (management console port); 2 USB (disabled in FIPS-CC mode except for power), 1 Micro USB Console (self-test output only in FIPS-CC mode), and Power input. Processor: Intel Atom C3436L (DP/MP) for PA-415 and PA-415-5G, and Intel Atom C3558R (DP/MP) for PA-445.
- PA-440/PA-450/PA-450R/PA-450R-5G/PA-455/PA-455-5G/PA-460: 8 RJ-45 10/100/100 ports for network traffic (Ethernet ports); 1 RJ-45 port to access the device GUI through an Ethernet interface (management ports); and 1 RJ-45 port for connecting a serial console (management console port); 2 USB (disabled in FIPS-CC mode except for power), 1 Micro USB Console (self-test output only in FIPS-CC mode), and Power input. Processor: Intel Atom C3558R (DP/MP) for 440, Intel Atom C3708 (DP/MP) for 450R, 450R-5G, and PA-455-5G, and Intel Atom C3758R (DP/MP) for 450, 455, and 460.

- PA-820: 4 RJ-45 10/100/1000 ports for network traffic (Ethernet ports); 8 Small Form-Factor Pluggable (SFP) Gbps ports for network traffic; 1 RJ-45 port to access the device GUI through an Ethernet interface (management ports); and 1 RJ-45 port for connecting a serial console (management console port); 1 USB (disabled in FIPS-CC mode except for power), 1 Micro USB Console (self-test output only in FIPS-CC mode), and 100-240V Power input. Processor: Cavium Octeon CN7240 MIPS64 (DP/MP)
- PA-850: 4 RJ-45 10/100/1000 ports for network traffic (Ethernet ports); 4/8 SFP; 0/4 SFP+ connectors for network traffic; 1 RJ-45 port to access the device GUI through an Ethernet interface (management ports); and 1 RJ-45 port for connecting a serial console (management console port); 1 USB (disabled in FIPS-CC mode except for power), 1 Micro USB Console (self-test output only in FIPS-CC mode), and 100-240V Power input. Processor: Cavium Octeon CN7240 MIPS64 (DP/MP)
- PA-1410/PA-1420: 12 RJ-45 10/100/1000 ports for network traffic. 10 Small Form-Factor Pluggable (SFP) ports for network traffic. 1 RJ-45 port to access the device management interfaces through an Ethernet interface. 1 RJ-45 port for connecting a serial console. 4 RJ-45 ports for high-availability (HA) control and synchronization, 1 USB (disabled in FIPS-CC mode except for power), 1 Micro USB Console (self-test output only in FIPS-CC mode), and 2 Power inputs. Processor: Intel Atom C5325 (PA-1410) and Intel Atom C5335C1 (PA-1420)
- PA-3220/PA-3250: 12 RJ-45 10/100/1000 ports for network traffic. 8 Small Form-Factor Pluggable (SFP) ports for network traffic. 1 RJ-45 port to access the device management interfaces through an Ethernet interface. 1 RJ-45 port for connecting a serial console. 2 RJ-45 ports for high-availability (HA) control and synchronization, 1 USB (disabled in FIPS-CC mode except for power), 1 Micro USB Console (self-test output only in FIPS-CC mode), and 2 Power inputs. Processor: Cavium Octeon CN7350 MIPS64 (DP) / Intel Pentium D1517 (MP)
- PA-3260: 12 RJ-45 10/100/1000 ports for network traffic. 8 Small Form-Factor Pluggable (SFP) ports for network traffic. 1 RJ-45 port to access the device management interfaces through an Ethernet interface. 1 RJ-45 port for connecting a serial console. 2 RJ-45 ports for high-availability (HA) control and synchronization , 1 USB (disabled in FIPS-CC mode except for power), 1 Micro USB Console (self-test output only in FIPS-CC mode), and 2 Power inputs. Processor: Cavium Octeon CN7360 MIPS64 (DP) / Intel Pentium D1517 (MP)
- PA-3410/PA-3420: 12 RJ-45 10/100/1000 ports for network traffic. 10 Small Form-Factor Pluggable (SFP) ports for network traffic. 1 RJ-45 port to access the device management interfaces through an Ethernet interface. 1 RJ-45 port for connecting a serial console. 4 RJ-45 ports for high-availability (HA) control and synchronization, 1 USB (disabled in FIPS-CC mode except for power), 1 Micro USB Console (self-test output only in FIPS-CC mode), and 2 Power inputs. Processor: Intel Atom 5332 (PA-3410) and Intel Atom 5342 (PA-3420)
- PA-3430/PA-3440: 12 RJ-45 10/100/1000 ports for network traffic. 10 Small Form-Factor Pluggable (SFP) ports for network traffic, and 2 zQSFP+ ports. 1 RJ-45 port to access the device management interfaces through an Ethernet interface. 1 RJ-45 port for connecting a serial console. 4 RJ-45 ports for high-availability (HA) control and synchronization, 1 USB (disabled in FIPS-CC mode except for power), 1 Micro USB Console (self-test output only in FIPS-CC mode), and 2 Power inputs. Processor: Intel Atom 5352 (PA-3430) and Intel Atom 5362 (PA-3440)
- PA-5220: 4 100/1000/10G Cu, 16 1G/10G SFP/SFP+, 4 40G QSFP+ for network traffic; 2 RJ-45 port to access the device management interfaces through an Ethernet interface; 1 RJ-45 port for connecting a serial console, 1 40G QSFP+ HA for high-availability (HA) control and synchronization, 1 USB (disabled in FIPS-CC mode except for power), and 2 100-240V Power inputs. Processor: Cavium Octeon CN7885 MIPS64 (DP) / Intel Xeon D1548 (MP)
- PA-5250: 4 100/1000/10G Cu, 16 1G/10G SFP/SFP+, 4 40G/100G QSFP28 for network traffic; 2 RJ-45 port to access the device management interfaces through an Ethernet interface; 1 RJ-45 port for connecting a serial console, 1 40G/100G QSFP28 for high-availability (HA) control and synchronization, 1 USB (disabled in FIPS-CC mode except for power), and 2 100-240V Power inputs. Processor: Cavium Octeon CN7890 MIPS64 (DP) / Intel Xeon D1567 (MP)

- PA-5260/PA-5280: 4 100/1000/10G Cu, 16 1G/10G SFP/SFP+, 4 40G/100G QSFP28 for network traffic; 2 RJ-45 port to access the device management interfaces through an Ethernet interface; 1 RJ-45 port for connecting a serial console, 1 40G/100G QSFP28 for high-availability (HA) control and synchronization, 1 USB (disabled in FIPS-CC mode except for power), and 2 100-240V Power inputs. Processor: Cavium Octeon CN7890 MIPS64 (DP) / Intel Xeon D1567 (MP)
- PA-5410/PA-5420/PA-5430/PA-5440/PA-5445: 8 100/1000/10G Cu, 12 1G/10G SFP/SFP+, 4 40G/100G QSFP28 for network traffic; 2 RJ-45 port to access the device management interfaces through an Ethernet interface; 1 RJ-45 port for connecting a serial console, 2 40G/100G QSFP28 for high-availability (HA) control and synchronization, 1 USB (disabled in FIPS-CC mode except for power), and 2 100-240V Power inputs. Processor: AMD EPYC 7352 (PA-5410), AMD EPYC 7452 (PA-5420), AMD EPYC 7642 (PA-5430), AMD EPYC 7742 (PA-5440), and AMD EPYC 7003 (PA-5445)
- PA-5450: For network connectivity, the PA-5450 requires at least one NC (PA-5400-NC-A). Each PA-5400-NC-A offers multiple connectivity ports as listed: 100/1000/10G Cu (4), 1G/10G SFP/SFP+ (12), and 40G/100G QSFP28 (2), and 2 Power inputs. For packet and security processing, the PA-5450 uses DPCs (PA-5400-DPC-A). The MPC (PAN-PA-5400-MPC-A) acts as a dedicated point of contact for controlling all aspects of the PA-5450 and has 2 HA ports, 1 RJ-45 Console port, 2 Management ports, 1 USB (disabled in FIPS-CC mode except for power) and 1 Micro USB Console (self-test output only in FIPS-CC mode). Processor: Intel Xeon D-2187NT (DP/MP)
- PA-7050: 12 gig copper ports for network traffic, 8 Small Form-Factor Pluggable (SFP) ports for network traffic and 4 SFP+ ports for network traffic per blade OR 2 Quad Small Form-Factor Pluggable (QSFP) for network traffic per blade and 12 SFP+ ports for network traffic per blade (6 blades max). 1 RJ-45 port to access the device management interfaces through an Ethernet interface. 1 RJ-45 port for connecting a serial console. 2 QSFP ports for high-availability (HA) control and synchronization, 1 USB (disabled in FIPS-CC mode except for power), and 4 100-240V Power inputs. Processor: Cavium Octeon CN7890 MIPS64 (DP) / Xeon D-1567 (MP)
- PA-7080: 12 gig copper ports for network traffic, 8 Small Form-Factor Pluggable (SFP) ports for network traffic and 4 SFP+ ports for network traffic per blade OR 2 Quad Small Form-Factor Pluggable (QSFP) for network traffic per blade and 12 SFP+ ports for network traffic per blade (10 blades max). 1 RJ-45 port to access the device management interfaces through an Ethernet interface. 1 RJ-45 port for connecting a serial console. 2 QSFP ports for high-availability (HA) control and synchronization, 1 USB (disabled in FIPS-CC mode except for power), and 8 100-240V Power inputs. Processor: Cavium Octeon CN7890 MIPS64 (DP) / Xeon D-1567 (MP)
- PA-7500: Chassis modular platform that supports Management Processing Card (MPC) [1 RJ-45 console port, 1 USB port, 2 SFP/SFP+/SFP management ports, 2 HSCI ports, and 2 SFP/SFP+ logging ports], Network Processing Card (NPC) [14 400Gbps and 100Gbps QSFP28 and QSFP+ ports, and 12 SFP-DD ports], and Data Processing Card (DPC) [no ports]. The Switch Fabric Card (SFC) provides data plane connectivity to the other interface cards and has one RJ-45 console ports. MPC and DPC – Intel D-2798NX, NPC – Intel Atom P5752, and SFC – Intel Atom C3758R.

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

- A computer securely 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).

Traffic logs, which record information about each traffic flow or problems with the network traffic, are logged locally by default. However, the product offers the capability to send the logs as SNMP traps, Syslog messages, or email notifications. Traffic logging and the use of email notifications and the SNMP and SMTP servers have not been subject to testing in the evaluated configuration.

The operational environment includes the following:

- Syslog server

- VPN gateway peer(s)
- Palo Alto Networks GlobalProtect application
- Workstation
    - Web browsers – Chrome (version 119 or later) browser.
    - SSHv2 client

The operational environment includes a domain controller and the User Identification Agent is installed on one or more PCs in the operational environment, and is supported on Windows Server 2016 and and Windows Server 2019.

Table 1 TOE Platforms

Product Identification	Illustration	Description
PA-410		• 1.7/1.3 Gbps Firewall throughput (HTTP/appmix) • 0.6/0.7 Gbps Threat Prevention throughput (HTTP/appmix) • 0.93 Gbps IPsec VPN throughput • 64,000 max sessions • 13,000 New sessions per second
PA-410R-5G		• 1.7/1.3 Gbps Firewall throughput (HTTP/appmix) • 0.6/0.7 Gbps Threat Prevention throughput (HTTP/appmix) • 0.93 Gbps IPsec VPN throughput • 64,000 max sessions • 13,000 New sessions per second
PA-415		• 1.65/1.2 Gbps Firewall throughput (HTTP/appmix) • 0.6/0.69 Gbps Threat Prevention throughput (HTTP/appmix) • 0.92 Gbps IPsec VPN throughput • 64,000 max sessions • 12,000 New sessions per second
PA-415-5G		• 1.65/1.2 Gbps Firewall throughput (HTTP/appmix) • 0.6/0.69 Gbps Threat Prevention throughput (HTTP/appmix) • 0.92 Gbps IPsec VPN throughput • 64,000 max sessions • 12,000 New sessions per second
PA-440		• 3.0/2.4 Gbps Firewall throughput (HTTP/appmix) • 0.9/1.0 Gbps Threat Prevention throughput (HTTP/appmix) • 1.6 Gbps IPsec VPN throughput • 200,000 max sessions • 39,000 New sessions per second
PA-445		• 2.8/2.2 Gbps Firewall throughput (HTTP/appmix) • 1.0/1.0 Gbps Threat Prevention throughput (HTTP/appmix)

Product Identification	Illustration	Description
		• 1.6 Gbps IPsec VPN throughput • 200,000 max sessions • 37,500 New sessions per second
PA-450		• 3.8/3.2 Gbps Firewall throughput (HTTP/appmix) • 1.6/1.7 Gbps Threat Prevention throughput (HTTP/appmix) • 2.2 Gbps IPsec VPN throughput • 300,000 max sessions • 52,000 New sessions per second
PA-450R		• 3.8/3.2 Gbps Firewall throughput (HTTP/appmix) • 1.6/1.7 Gbps Threat Prevention throughput (HTTP/appmix) • 2.2 Gbps IPsec VPN throughput • 300,000 max sessions • 52,000 New sessions per second
PA-450R-5G		• 3.8/3.2 Gbps Firewall throughput (HTTP/appmix) • 1.6/1.7 Gbps Threat Prevention throughput (HTTP/appmix) • 2.2 Gbps IPsec VPN throughput • 300,000 max sessions • 52,000 New sessions per second
PA-455		• 3.2/3.3 Gbps Firewall throughput (HTTP/appmix) • 1.6/2 Gbps Threat Prevention throughput (HTTP/appmix) • 3.1 Gbps IPsec VPN throughput • 300,000 max sessions • 74,000 New sessions per second
PA-455-5G		• 3.6 Gbps Firewall throughput (HTTP/appmix) • 2.3 Gbps Threat Prevention throughput (HTTP/appmix) • 2 Gbps IPsec VPN throughput • 300,000 max sessions • 55,000 New sessions per second
PA-460		• 5.2/4.7 Gbps Firewall throughput (HTTP/appmix) • 2.4/2.6 Gbps Threat Prevention throughput (HTTP/appmix) • 3.1 Gbps IPsec VPN throughput • 400,000 max sessions • 74,000 New sessions per second
PA-820		• 1.9 Gbps firewall throughput (App-ID enabled) • 780 Mbps threat prevention throughput • 500 Mbps IPsec VPN throughput • 192,000 max sessions • 9,500 new sessions per second • 1000 IPsec VPN tunnels/tunnel interfaces

Product Identification	Illustration	Description
		• 5 virtual routers • 40 security zones • 1,500 max number of policies
PA-850		• 1.9 Gbps firewall throughput (App-ID enabled) • 780 Mbps threat prevention throughput • 500 Mbps IPsec VPN throughput • 192,000 max sessions • 9,500 new sessions per second • 1000 IPsec VPN tunnels/tunnel interfaces • 5 virtual routers • 40 security zones • 1,500 max number of policies
PA-1410		• 8.9/6.8 Gbps firewall throughput (HTTP/appmix) • 3.3/3.2 Mbps threat prevention throughput (HTTP/appmix) • 4.6 Gbps IPsec VPN throughput • 945,000 max sessions • 100,000 new sessions per second
PA-1420		• 9.9/9.5 Gbps firewall throughput (HTTP/appmix) • 5.0/4.8 Mbps threat prevention throughput (HTTP/appmix) • 6.5 Gbps IPsec VPN throughput • 1,400,000 max sessions • 140,000 new sessions per second
PA-3220		• 4.6/4.6 Gbps firewall throughput (App-ID enabled) • 2.2/2.6 Gbps Threat Prevention throughput • 2.5 Gbps IPsec VPN throughput • 1,000,000 max sessions • 57,000 new sessions per second • 4,000 IPsec VPN tunnels/tunnel interfaces • 1,024 SSL VPN Users • 10 virtual routers • 1/6 virtual systems (base/max) • 200 security zones • 2,500 max number of policies
PA-3250		• 6/7 Gbps firewall throughput (App-ID enabled) • 2.6/3.1 Gbps Threat Prevention throughput • 3.2 Gbps IPsec VPN throughput • 2,000,000 max sessions • 84,000 new sessions per second • 6,000 IPsec VPN tunnels/tunnel interfaces • 2,048 SSL VPN Users • 10 virtual routers • 1/6 virtual systems (base/max)

Product Identification	Illustration	Description
		• 200 security zones • 5,000 max number of policies
PA-3260		• 8.4/10 Gbps firewall throughput (App-ID enabled) • 3.9/4.7 Gbps Threat Prevention throughput • 4.8 Gbps IPsec VPN throughput • 3,000,000 max sessions • 118,000 new sessions per second • 6,000 IPsec VPN tunnels/tunnel interfaces • 2,048 SSL VPN Users • 10 virtual routers • 1/6 virtual systems (base/max) • 200 security zones • 5,000 max number of policies
PA-3410		• 14.5/11.6 Gbps Firewall throughput (HTTP/appmix) • 5.2/5.9 Gbps Threat Prevention throughput (HTTP/appmix) • 6.8 Gbps IPsec VPN throughput • 1.4 M max sessions • 145,000 new sessions per second • 1/11 virtual systems (base/max)
PA-3420		• 20.8/16.9 Gbps Firewall throughput (HTTP/appmix) • 7.6/8.7 Gbps Threat Prevention throughput (HTTP/appmix) • 9.9 Gbps IPsec VPN throughput • 2 M max sessions • 205,000 new sessions per second • 1/11 virtual systems (base/max)
PA-3430		• 25.5/20.5 Gbps Firewall throughput (HTTP/appmix) • 9.2/10.5 Gbps Threat Prevention throughput (HTTP/appmix) • 12.2 Gbps IPsec VPN throughput • 2.5 M max sessions • 240,000 new sessions per second • 1/11 virtual systems (base/max)
PA-3440		• 30.2/24 Gbps Firewall throughput (HTTP/appmix) • 11/12.8 Gbps Threat Prevention throughput (HTTP/appmix) • 14.5 Gbps IPsec VPN throughput • 3 M max sessions • 268,000 new sessions per second • 1/11 virtual systems (base/max)
PA-5220		• 17/20 Gbps firewall throughput (HTTP/appmix) • 8/9 Gbps Threat Prevention throughput (HTTP/appmix) • 8 Gbps IPsec VPN throughput • 4,000,000 max sessions

Product Identification	Illustration	Description
		• 150,000 New sessions per second • 20 virtual routers • 10/20 Virtual systems (base/max)
PA-5250		• 39/40 Gbps firewall throughput (HTTP/appmix) • 18/23 Gbps Threat Prevention throughput (HTTP/appmix) • 16 Gbps IPsec VPN throughput • 8,000,000 max sessions • 284,000 New sessions per second • 125 virtual routers • 25/125 Virtual systems (base/max)
PA-5260		• 60/67 Gbps Firewall throughput (HTTP/appmix) • 28/33 Gbps Threat Prevention throughput (HTTP/appmix) • 24 Gbps IPsec VPN throughput • 32,000,000 max sessions • 390,000 New sessions per second • 225 virtual routers • 25/225 Virtual systems (base/max)
PA-5280		• 60/67 Gbps Firewall throughput (HTTP/appmix) • 28/33 Gbps Threat Prevention throughput (HTTP/appmix) • 24 Gbps IPsec VPN throughput • 64,000,000 max sessions • 390,000 New sessions per second • 225 virtual routers • 25/225 Virtual systems (base/max)
PA-5410		• 45.2/36.7 Gbps Firewall throughput (HTTP/appmix) • 22/23.5 Gbps Threat Prevention throughput (HTTP/appmix) • 21 Gbps IPsec VPN throughput • 4.1 M max sessions • 246,000 new sessions per second • 10/20 virtual systems (base/max)
PA-5420		• 53.7/47.5 Gbps Firewall throughput (HTTP/appmix) • 28.8/30.5 Gbps Threat Prevention throughput (HTTP/appmix) • 28.7 Gbps IPsec VPN throughput • 6.2 M max sessions • 315,000 new sessions per second • 15/65 virtual systems (base/max)

Product Identification	Illustration	Description
PA-5430		• 63/59.4 Gbps Firewall throughput (HTTP/appmix) • 37.6/40.9 Gbps Threat Prevention throughput (HTTP/appmix) • 42 Gbps IPsec VPN throughput • 8.3 M max sessions • 366,000 new sessions per second • 25/125 virtual systems (base/max)
PA-5440		• 93.5/72 Gbps Firewall throughput (HTTP/appmix) • 61.5/52 Gbps Threat Prevention throughput (HTTP/appmix) • 58 Gbps IPsec VPN throughput • 20 M max sessions • 390,000 new sessions per second • 25/225 virtual systems (base/max)
PA-5445		• 93.5/90 Gbps Firewall throughput (HTTP/appmix) • 71.5/72 Gbps Threat Prevention throughput (HTTP/appmix) • 64 Gbps IPsec VPN throughput • 48 M max sessions • 449,000 new sessions per second • 25/225 virtual systems (base/max)
PA-5450		• 200/200 Gbps Firewall throughput (HTTP/appmix) • 120/148 Gbps Threat Prevention throughput (HTTP/appmix) • 79 Gbps IPsec VPN throughput • 100,000,000 max sessions • 3.5M New sessions per second • 25/225 Virtual systems (base/max)
PA-7050		• 380/430 Gbps firewall throughput • 366 Gbps Threat Prevention throughput (DSRI enabled) • 176/210 Gbps Threat Prevention throughput • 144 Gbps IPsec VPN throughput • 192 M max sessions • 2.9 M new sessions per second • 25/225 virtual systems (base/max)
PA-7080		• 630/720 Gbps firewall throughput • 610 Gbps Threat Prevention throughput (DSRI enabled) • 294/350 Gbps Threat Prevention throughput • 240 Gbps IPsec VPN throughput

Product Identification	Illustration	Description
		• 320 M max sessions • 4.8 M new sessions per second • 25/225 virtual systems (base/max)
PA-7500		• 310/1,500* Gbps firewall throughput • 250/1,440 Gbps Threat Prevention throughput • 67/407 Gbps IPsec VPN throughput • 73M/440M max sessions • 1.2/7.2 M new sessions per second • 25/225 virtual systems (base/max) * - Maximum results based on configuration using six PA-7500-DPC-A cards and two PA-7500-NPC-A cards.
Virtual Appliances		
VM-50		• 50,000 max sessions • 250 security rules • 1,000 dynamic IP addresses • 15 Security zones • 250 IPsec VPN tunnels • 250 TLS VPN tunnels
VM-100		• 250,000 max sessions • 1,500 security rules • 2,500 dynamic IP addresses • 40 Security zones • 1,000 IPsec VPN tunnels • 500 TLS VPN tunnels
VM-300		• 800,000 max sessions • 10,000 security rules • 100,000 dynamic IP addresses • 40 Security zones • 2,000 IPsec VPN tunnels • 2,000 TLS VPN tunnels
VM-500		• 2,000,000 max sessions • 10,000 security rules • 100,000 dynamic IP addresses • 200 Security zones • 4,000 IPsec VPN tunnels • 6,000 TLS VPN tunnels

Product Identification	Illustration	Description
VM-700		• 10, 000,000 max sessions • 20,000 security rules • 100000 dynamic IP addresses • 200 Security zones • 8,000 IPsec VPN tunnels • 12,000 TLS VPN tunnels

2.2.2 Logical Boundaries

This section summarizes the security functions provided by the TOE:

- Security Audit
- Cryptographic Support
- User Data Protection
- Identification and Authentication
- Security Management
- Protection of the TSF
- TOE Access
- Trusted Path/Channels
- Stateful Traffic Filtering
- Packet Filtering
- Intrusion Prevention System

2.2.2.1 Security Audit

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

2.2.2.2 Cryptographic Support

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

2.2.2.3 User Data Protection

The TOE is designed to ensure that it does not inadvertently reuse data found in network traffic.

2.2.2.4 Identification and Authentication

The TOE requires all users accessing the TOE user interfaces to be successfully identified and authenticated before they can access any security management functions available in the TOE. The TOE offers network accessible (HTTPS, SSH, IPsec) connections to the GUI, 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 or public-key, and role (set of privileges), which it uses to authenticate the user and to associate that user with an authorized role. In addition, the TOE can authenticate users using X.509v3 certificates and can be configured to lock a user out after a configurable number of unsuccessful authentication attempts.

2.2.2.5 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/API/CLI using an HTTPS/TLS, IPsec, 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 login banner, configure the idle timeout, configure IKE/IPsec VPN gateways, configure threat signature rules, and other management functions. 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.

2.2.2.6 Protection of the TSF

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

It protects particularly sensitive data such as stored passwords and cryptographic keys so that they are not accessible even by an administrator. It also provides 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 and transition to a secure, maintenance state. It also includes a mechanism to verify TOE updates to prevent malicious or other unexpected changes in the TOE.

2.2.2.7 TOE Access

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

2.2.2.8 Trusted Path/Channels

The TOE protects interactive communication with remote administrators using SSH, HTTP over TLS (HTTPS), or IPsec. SSH, TLS, and IPsec ensure both integrity and disclosure protection. Note: HTTPS traffic can be tunneled through IPsec secure channel.

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

2.2.2.9 Stateful Traffic Filtering

The TOE provides a stateful traffic filter firewall for layers 3 and 4 (IP and TCP/UDP) network traffic optimized through the use of stateful packet inspection.

An administrator can configure the TOE to control the type of information that is allowed to pass through the TOE. The administrator defines the security zone and applies security policies to network traffic attempting to traverse the TOE to determine what actions to take.

The TOE groups interfaces into security zones. Each zone identifies one or more interfaces on the TOE. Separate zones must be created for each type of interface (Layer 2, Layer 3, or virtual wire), and each interface must be assigned to a zone before it can process traffic. Security policies provide the firewall rule sets that specify whether to block or allow network connections, based on the source and destination zones, and addresses, and the application service (such as UDP port 67 or TCP port 80). Security policy rules are processed in sequence, applying the first rule that matches the incoming traffic.

2.2.2.10 Packet Filtering

The TOE provides packet filtering and secure IPsec tunneling. The tunnels can be established between two trusted VPN peers as well as between remote VPN clients and the TOE. An administrator can configure security policies that determine whether to block, allow, or log a session based on traffic attributes such as the source and destination security zone, the source and destination IP address, the application, user, and the service.

2.2.2.11 Intrusion Prevention System

The TOE provides IPS functionalities such as malicious list blocking, reconnaissance and Denial of Service (DoS) flooding protection, anomaly-based and signature-based traffic detection and response mechanisms.

2.3 TOE Documentation

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

For PAN-OS v11.2, these documents are found here:

- https://docs.paloaltonetworks.com/pan-os/11-2/pan-os-release-notes/related-documentation/related-documentation-for-pan-os-11-2

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, IKE/IPsec. The features below and Normal mode 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, IPsec, 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 are only allowed for pre-operational installation, configuration, and post-operational maintenance and troubleshooting.
TLS and SSH Decryption Policies	The TLS and SSH decryption policies are not evaluated and therefore, these features are out of scope.
File Blocking, DLP, and URL Filtering Security Policies	The File Blocking, DLP (Data Loss Prevention), and URL Filtering security policies/profiles are not evaluated and therefore, these features are out of scope.
API request over HTTP	By default, the TOE supports API requests over HTTPS or HTTPS tunneled over IPsec. API request over HTTP is disabled and cannot be enabled in the evaluated configuration.
Any features not associated with SFRs in claimed [NDcPP], [FW-Module], [VPNGW-Module], and [IPS-Module]	NDcPP forbids adding additional requirements to the Security Target (ST). If additional functionalities or products are mentioned in the ST, it is for completeness only.

3. Security Problem Definition

This security target includes by reference the Security Problem Definition (composed of organizational policies, threat statements, and assumption) from [NDcPP], [FW-Module], [VPNGW-Module], and [IPS-Module].

In general, the [NDcPP], [FW-Module], [VPNGW-Module], and [IPS-Module] have 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. NOTE: A.COMPONENTS_RUNNING is not applicable because this is not a distributed TOE.

4. Security Objectives

Like the Security Problem Definition, this security target includes by reference the Security Objectives from the [NDcPP], [FW-Module], [VPNGW-Module], and [IPS-Module]. 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], [FW-Module], [VPNGW-Module], and [IPS-Module] has presented Security Objectives appropriate for network infrastructure devices, such as is applicable to the Palo Alto TOE. NOTE: OE.COMPONENTS_RUNNING is not applicable because this is not a distributed TOE.

4.1 Security Objectives for the Operational Environment

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

	platform on which the VM runs is removed from its operational environment.
OE.CONNECTIONS	The TOE is connected to distinct networks in a manner that ensures that the TOE security policies will be enforced on all applicable network traffic flowing among the attached networks. [VPNGW-Module] TOE administrators will ensure that the TOE is installed in a manner that will allow the TOE to effectively enforce its policies on network traffic of monitored networks. [IPS-Module]
OE.VM_CONFIGURATION	For vNDs, the Security Administrator ensures that the VS and VMs are configured to ● reduce the attack surface of VMs as much as possible while supporting ND functionality (e.g., remove unnecessary virtual hardware, turn off unused inter-VM communications mechanisms), and ● correctly implement ND functionality (e.g., ensure virtual networking is properly configured to support network traffic, management channels, and audit reporting). The VS should be operated in a manner that reduces the likelihood that vND operations are adversely affected by unauthorizedfeatures such as cloning, save/restore, suspend/resume, and live migration. If possible, the VS should be configured to make use of features that leverage the VS's privileged position to provide additional security functionality. Such features could include malware detection through VM introspection, measured VM boot, or VM snapshot for forensic analysis.

5. IT Security Requirements

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

The SFRs have all been drawn from the [NDcPP], [FW-Module], [VPNGW-Module], [SSHPKG], and [IPS-Module].

The SARs are the set of SARs specified in the PP-Configuration for Network Devices, Stateful Traffic Filter Firewalls, Virtual Private Network (VPN) Gateways, and Intrusion Prevention Systems.

5.1 Extended Requirements

All the extended requirements in this ST have been drawn from the [NDcPP], [FW-Module], [VPNGW-Module], [SSHPKG], and [IPS-Module]. The [NDcPP], [FW-Module], [VPNGW-Module], [SSHPKG], and [IPS-Module] define all the extended SFRs (*_EXT) and since they are not redefined in this ST, those PPs and Modules should be consulted for more information regarding those CC extensions.

5.2 TOE Security Functional Requirements

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

Table 3 TOE Security Functional Components

Requirement Class	Requirement Component	Location
FAU: Security Audit	FAU_GEN.1: Audit Data Generation	NDcPP
	FAU_GEN.1/VPN: Audit Data Generation (VPN Gateway)	VPNGW
	FAU_GEN.1/IPS: Audit Data Generation (IPS)	IPS
	FAU_GEN.2: User Identity Association	NDcPP
	FAU_STG_EXT.1: Protected Audit Event Storage	NDcPP
	FAU_STG.1: Protected Audit Trail Storage	NDcPP
FCS: Cryptographic Support	FCS_CKM.1: Cryptographic Key Generation	NDcPP
	FCS_CKM.1/IKE: Cryptographic Key Generation (for IKE Peer Authentication)	VPNGW
	FCS_CKM.2: Cryptographic Key Establishment	NDcPP
	FCS_CKM.4: Cryptographic Key Destruction	NDcPP
	FCS_COP.1/DataEncryption: Cryptographic Operation (AES Data Encryption/Decryption)	NDcPP
	FCS_COP.1/SigGen: Cryptographic Operation (Signature Generation and Verification)	NDcPP
	FCS_COP.1/Hash: Cryptographic Operation (Hash Algorithm)	NDcPP
	FCS_COP.1/KeyedHash: Cryptographic Operation (Keyed Hash Algorithm)	NDcPP
	FCS_RBG_EXT.1: Random Bit Generation	NDcPP

Requirement Class	Requirement Component	Location
	FCS_HTTPS_EXT.1: HTTPS Protocol	NDcPP
	FCS_SSH_EXT.1: SSH Protocol	SSHPKG
	FCS_SSHS_EXT.1: SSH Protocol – Server	SSHPKG
	FCS_TLSC_EXT.1: TLS Client Protocol	NDcPP
	FCS_TLSS_EXT.1: TLS Server Protocol	NDcPP
	FCS_IPSEC_EXT.1: IPsec Protocol	NDcPP
FDP: User Data Protection	FDP_RIP.2: Full Residual Information Protection	FW
FIA: Identification and Authentication	FIA_AFL.1: Authentication Failure Management	NDcPP
	FIA_PMG_EXT.1: Password Management	NDcPP
	FIA_PSK_EXT.1: Pre-Shared Key Composition	NDcPP
	FIA_PSK_EXT.2: Generated Pre-Shared Keys	NDcPP
	FIA_UIA_EXT.1: User Identification and Authentication	NDcPP
	FIA_X509_EXT.1/Rev: X.509 Certificate Validation	NDcPP
	FIA_X509_EXT.2: X.509 Certificate Authentication	NDcPP
	FIA_X509_EXT.3: X.509 Certificate Requests	NDcPP
FMT: Security Management	FMT_MOF.1/ManualUpdate: Management of Security Functions Behaviour	NDcPP
	FMT_MOF.1/Services: Management of Security Functions Behaviour	NDcPP
	FMT_MTD.1/CryptoKeys: Management of TSF Data	NDcPP
	FMT_MTD.1/CoreData: Management of TSF Data	NDcPP
	FMT_SMF.1: Specification of Management Functions	NDcPP
	FMT_SMF.1/FFW: Specification of Management Functions	FW
	FMT_SMF.1/VPN: Specification of Management Functions	VPNGW
	FMT_SMF.1/IPS: Specification of Management Functions (IPS)	IPS
	FMT_SMR.2: Restrictions on Security Roles	NDcPP
FPT: Protection of the TSF	FPT_SKP_EXT.1: Protection of TSF Data (for reading of all symmetric keys)	NDcPP
	FPT_APW_EXT.1: Protection of Administrator Passwords	NDcPP
	FPT_FLS.1/SelfTest Fail Secure (Self-test Failures)	VPNGW
	FPT_STM_EXT.1: Reliable Time Stamps	NDcPP

Requirement Class	Requirement Component	Location
	FPT_TST_EXT.1: TSF Testing	NDcPP
	FPT_TST_EXT.3: Self-Test with Defined Methods	VPNGW
	FPT_TUD_EXT.1: Trusted Update	NDcPP
FTA: TOE Access	FTA_SSL.3: TSF-initiated Termination	NDcPP
	FTA_SSL.4: User-initiated Termination	NDcPP
	FTA_TAB.1: Default TOE Access Banners	NDcPP
FTP: Trusted Path/Channels	FTP_ITC.1: Inter-TSF Trusted channel	NDcPP
	FTP_ITC.1/VPN: Inter-TSF Trusted channel (VPN Communications)	VPNGW
	FTP_TRP.1/Admin: Trusted Path	NDcPP
FFW: Stateful Traffic Filter Firewall	FFW_RUL_EXT.1: Stateful Traffic Filtering	FW
	FFW_RUL_EXT.2: Stateful Filtering of Dynamic Protocols	FW
FPF: Packet Filtering	FPF_RUL_EXT.1: Rules for Packet Filtering	VPNGW
IPS: Intrusion Prevention	IPS_ABD_EXT.1: Anomaly-Based IPS Functionality	IPS
	IPS_IPB_EXT.1: IP Blocking	IPS
	IPS_NTA_EXT.1: Network Traffic Analysis	IPS
	IPS_SBD_EXT.1: Signature-Based IPS Functionality	IPS

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 not specified level of audit; and
c) *All administrative actions comprising:*
- *Administrative login and logout (name of Administrator account shall be logged if individual accounts are required for Administrators).*
- *Changes to TSF data related to configuration changes (in addition to the information that a change occurred it shall be logged what has been changed).*
- *Generating/import of, changing, or deleting of cryptographic keys (in addition to the action itself a unique key name or key reference shall be logged).*
- *[**Resetting passwords (name of related user account shall be logged)**];*

d) *Specifically defined auditable events listed in Table **4***

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

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

Table 4 Auditable Events (Network Device, Firewall)

Requirement	Auditable Events	Additional Audit Record Contents
FAU_GEN.1	None.	None.
FAU_GEN.2	None.	None.
FAU_STG_EXT.1	Configuration of local audit settings.	Identity of account making changes to the audit configuration.
FAU_STG.1	None.	None.
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_SSH_EXT.1	• Failure to establish a SSH session. • Establishment of SSH connection • Termination of SSH connection session	• Reason for failure. • Non-TOE endpoint of connection (IP Address)
	• None	• None
FCS_TLSC_EXT.1	Failure to establish a TLS session.	Reason for failure
FCS_TLSS_EXT.1	Failure to establish a TLS session.	Reason for failure
FCS_IPSEC_EXT.1	Failure to establish an IPsec SA	Reason for failure
FDP_RIP.2	None.	None.
FIA_AFL.1	Unsuccessful login attempts limit is met or exceeded.	Origin of the attempt (e.g., IP address)
FIA_PMG_EXT.1	None.	None.
FIA_UIA_EXT.1	All use of identification and authentication mechanisms.	Origin of the attempt (e.g., IP address).
FIA_X509_EXT.1/Rev	• Unsuccessful attempt to validate a certificate • Any addition, replacement or removal of trust anchors[4] in the TOE's trust store	• Reason for failure of certificate validation • Identification of certificates added, replaced or removed

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

Requirement	Auditable Events	Additional Audit Record Contents
		as trust anchor in the TOE's trust store
FIA_X509_EXT.2	None.	None.
FIA_X509_EXT.3	None.	None.
FIA_PSK_EXT.1	None.	None.
FIA_PSK_EXT.2	None.	None.
FMT_MOF.1/ManualUpdate	Any attempt to initiate a manual update	None.
FMT_MOF.1/Services	None.	None.
FMT_MTD.1/CryptoKeys	None.	None.
FMT_MTD.1/CoreData	None.	None.
FMT_SMF.1	All management activities of TSF data	None.
FMT_SMF.1/FFW	All management activities of TSF data (including creation, modification, and deletion of firewall rules).	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.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.	• None • None • Reason for failure

Requirement	Auditable Events	Additional Audit Record Contents
	• Failure of the trusted channel functions.	
FTP_TRP.1/Admin	• Initiation of the trusted path. • Termination of the trusted path. • Failures of the trusted path functions.	• None • None • Reason for failure
FFW_RUL_EXT.1	Application of rules configured with the 'log' operation	Source and destination addresses Source and destination ports Transport Layer Protocol TOE Interface
FFW_RUL_EXT.2	Dynamical definition of rule, Establishment of a session	None

***Application Note**: This SFR is defined in [NDcPP] but was modified per guidance in [FW-Module].*

FAU_GEN.1/VPN – Audit Data Generation (VPN Gateway)

FAU_GEN.1.1/VPN 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) Indication that TSF self-test was completed
c) Failure of self-test
d) All auditable events for the [*not specified*] level of audit; and
e) [*auditable events defined in **Table 5***]

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

a) Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and
b) For each audit event type, based on the auditable event definitions of the functional components included in the cPP/ST, [*additional information defined in **Table** 5*].

Table 5 Auditable Events (VPN Gateway)

Requirement	Auditable Events	Additional Audit Record Contents
FAU_GEN.1/VPN	No events specified.	N/A
FCS_CKM.1/IKE	No events specified.	N/A
FMT_SMF.1/VPN	All administrative actions	No additional information.
FPF_RUL_EXT.1	Application of rules configured with the 'log' operation	Source and destination addresses Source and destination ports Transport layer protocol

Requirement	Auditable Events	Additional Audit Record Contents
FPT_FLS.1/SelfTest	No events specified.	N/A
FPT_TST_EXT.3	No events specified.	N/A
FTP_ITC.1/VPN	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 channel establishment attempt.

FAU_GEN.1/IPS – Audit Data Generation (IPS)

FAU_GEN.1.1/IPS The TSF shall be able to generate an IPS audit record of the following auditable events:

a) Start-up and shut-down of the **IPS** functions;
b) All **IPS** auditable events for the [*not specified*] level of audit; and
c) [*All dissimilar IPS events;*
d) *All dissimilar IPS reactions;*
e) *Totals of similar events occurring within a specified time period;*
f) *Totals of similar reactions occurring within a specified time period;*
g) *The events in the IPS Events table.*
h) [***no other auditable events***]].

FAU_GEN.1.2/IPS The TSF shall record within each **IPS auditable event** record at least the following information:

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

Table 6 Auditable Events (IPS)

Requirement	Auditable Events	Additional Audit Record Contents
FMT_SMF.1/IPS	Modification of an IPS policy element.	Identifier or name of the modified IPS policy element (e.g. which signature, baseline, or known-good/known-bad list was modified)
IPS_ABD_EXT.1	Inspected traffic matches an anomaly-based IPS policy.	Source and destination IP addresses. The content of the header fields that were determined to match the policy. TOE interface that received the packets. Aspect of the anomaly-based IPS policy rule that triggered the event

Requirement	Auditable Events	Additional Audit Record Contents
		(e.g. throughput, time of day, frequency, etc.). Network-based action by the TOE (e.g., allowed, blocked, sent reset to source IP, sent blocking notification to firewall)
IPS_IPB_EXT.1	Inspected traffic matches a list of known-good or known-bad addresses applied to an IPS policy.	Source and destination IP addresses (and, if applicable, indication of whether the source and/or destination address matched the list). TOE interface that received the packet. Network-based action by the TOE (e.g. allowed, blocked, sent reset).
IPS_NTA_EXT.1	Modification of which IPS policies are active on a TOE interface. Enabling/disabling a TOE interface with IPS policies applied. Modification of which mode(s) is/are active on a TOE interface.	Identification of the TOE interface. The IPS policy and interface mode (if applicable).
IPS_SBD_EXT.1	Inspected traffic matches a signature-based IPS rule with logging enabled.	Name or identifier of the matched signature. Source and destination IP addresses. The content of the header fields that were determined to match the signature. TOE interface that received the packet. Network-based action by the TOE (e.g. allowed, blocked, sent reset).

FAU_GEN.2 – User Identity Association

FAU_GEN.2.1 For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.

FAU_STG_EXT.1 – Protected Audit Event Storage

FAU_STG_EXT.1.1 The TSF shall be able to transmit the generated audit data to an external IT entity using a trusted channel according to FTP_ITC.1.

FAU_STG_EXT.1.2 The TSF shall be able to store generated audit data on the TOE itself. In addition [

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

FAU_STG_EXT.1.3 The TSF shall maintain a [***log file***] of audit records in the event that an interruption of communication with the remote audit server occurs.

FAU_STG_EXT.1.4 The TSF shall be able to store [***persistent***] audit records locally with a minimum storage size of [***45 MB***].

FAU_STG_EXT.1.5 The TSF shall [***overwrite previous audit records according to the following rule: [overwrite oldest records first]***] when the local storage space for audit data is full.

FAU_STG_EXT.1.6 The TSF shall provide the following mechanisms for administrative access to locally stored audit records [***ability to view locally***].

FAU_STG.1 – Protected Audit Trail Storage

FAU_STG.1.1 The TSF shall protect the stored audit records in the audit trail from unauthorized deletion.

FAU_STG.2.1 The TSF shall be able to prevent unauthorized modifications to the stored audit records in the audit trail.

5.2.2 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] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4;***
- ***FFC Schemes using 'safe-prime' groups that meet the following: "NIST Special Publication 800-56A Revision 3, Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" and [RFC 3526].***

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

FCS_CKM.1/IKE – Cryptographic Key Generation (for IKE Peer Authentication)

FCS_CKM.1.1/IKE The TSF shall generate **asymmetric** cryptographic keys **used for IKE peer authentication** in accordance with a specified cryptographic key generation algorithm: [

- ***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 ECDSA schemes and implementing "NIST curves" P-384 and [P-256, P-521]]***

and [*no other key generation algorithm*]

and specified cryptographic key sizes [*equivalent to, or greater than, a symmetric key strength of 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: [

- ***Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography";***
- ***FFC Schemes using "safe-prime" groups that meet the following: NIST Special Publication 800-56A Revision 3, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography" and [groups listed in RFC 3526].***

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

FCS_CKM.4 – Cryptographic Key Destruction

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

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

that meets the following: *No Standard*.

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

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

***Application Note**: This SFR is defined in [NDcPP] but was modified per guidance in [VPNGW-Module].*

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

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

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

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

] and cryptographic key sizes [

- ***For RSA: modulus 2048 bits or greater,***
- ***For ECDSA: 256 bits or greater***

] 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 RSASSA-PKCS1v1_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***, ***implicit***] 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] platform-based noise source***] with a minimum of [***256 bits***] of entropy at least equal to the greatest security strength, according to ISO/IEC 18031:2011 Table C.1 "Security Strength Table for Hash Functions", of the keys and hashes that it will generate.

FCS_HTTPS_EXT.1 – HTTPS Protocol

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

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

FCS_SSH_EXT.1 – SSH Protocol

FCS_SSH_EXT.1.1 The TSF shall implement *SSH* acting as a [***server***] in accordance with that complies with RFCs 4251, 4252, 4253, 4254, [***4256***, ***4344***, ***5647***, ***5656***, ***6668***, ***8308***, ***8332***] and [*no other standard*].

FCS_SSH_EXT.1.2 The TSF shall ensure that the SSH protocol implementation supports the following user authentication methods: [

- ***"password" (RFC 4252),***
- ***"keyboard-interactive" (RFC 4256)***
- ***"publickey" (RFC 4252): [***
  - ***ssh-rsa (RFC 4253),***
  - ***rsa-sha2-256 (RFC 8332),***
  - ***rsa-sha2-512 (RFC 8332)***

  ]

] and no other methods.

FCS_SSH_EXT.1.3 The TSF shall ensure that, as described in RFC 4253, packets greater than [***262,105***] bytes in an SSH transport connection are dropped.

FCS_SSH_EXT.1.4 The TSF shall protect data in transit from unauthorized disclosure using the following mechanisms: [

- ***aes128-cbc (RFC 4253),***
- ***aes256-cbc (RFC 4253),***
- ***aes128-ctr (RFC 4344),***
- ***aes256-ctr (RFC 4344),***
- ***aes128-gcm@openssh.com (RFC 5647),***
- ***aes256-gcm@openssh.com (RFC 5647)***

] and no other mechanisms.

FCS_SSH_EXT.1.5 The TSF shall protect data in transit from modification, deletion, and insertion using: [

- ***hmac-sha2-256 (RFC 6668),***
- ***hmac-sha2-512 (RFC 6668),***
- ***implicit***

] and no other mechanisms.

FCS_SSH_EXT.1.6 The TSF shall establish a shared secret with its peer using: [

- ***ecdh-sha2-nistp256 (RFC 5656),***
- ***ecdh-sha2-nistp384 (RFC 5656),***
- ***ecdh-sha2-nistp521 (RFC 5656)***

] and no other mechanisms.

FCS_SSH_EXT.1.7 The TSF shall use *SSH KDF* as defined in [

- ***RFC 5656 (Section 4),***

] to derive the following cryptographic keys from a shared secret: *session keys*.

FCS_SSH_EXT.1.8 The TSF shall ensure that [

- ***a rekey of the session keys***

] occurs when any of the following thresholds are met:

- one hour connection time
- no more than one gigabyte of transmitted data, or
- no more than one gigabyte of received data.

FCS_SSHS_EXT.1 – SSH Protocol – Server

FCS_SSHS_EXT.1.1 The TSF shall authenticate itself to its peer (SSH Client) using: [

- ***ssh-rsa (RFC 4253),***
- ***rsa-sha2-256 (RFC 8332),***
- ***rsa-sha2-512 (RFC 8332),***
- ***ecdsa-sha2-nistp256 (RFC 5656),***
- ***ecdsa-sha2-nistp384 (RFC 5656),***

- ***ecdsa-sha2-nistp521 (RFC 5656)***

].

FCS_TLSC_EXT.1 – TLS Client Protocol without Mutual Authentication

FCS_TLSC_EXT.1.1 The TSF shall implement [***TLS 1.2 (RFC 5246), TLS 1.3 (RFC 8446)***] supporting 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_AES_128_GCM_SHA256***
- ***TLS_AES_256_GCM_SHA384***

] and no other ciphersuites.

FCS_TLSC_EXT.1.2 The TSF shall verify that the presented identifier matches [***the reference identifier per RFC 6125 Section 6***] and no other attribute types].

FCS_TLSC_EXT.1.3 The TSF shall not establish a trusted channel if the server certificate is invalid [

- ***without any administrator override mechanism***].

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

FCS_TLSC_EXT.1.5 The TSF shall [

- ***present the signature_algorithms extension with support for the following algorithms: [***
  - ***rsa_pkcs1 with sha256(0x0401),***
  - ***rsa_pkcs1with sha384(0x0501),***
  - ***rsa_pkcs1 with sha512(0x0601),***
  - ***ecdsa_secp256r1 with sha256(0x0403),***
  - ***ecdsa_secp384r1 with sha384(0x0503),***
  - ***ecdsa_secp521r1 with sha512(0x0603),***
  - ***rsa_pss_rsae with sha256(0x0804),***
  - ***rsa_pss_rsae with sha384(0x0805),***
  - ***rsa_pss_rsae with sha512(0x0806),***
  - ***rsa_pss_pss with sha256(0x0809),***

- ***rsa_pss_pss with sha384(0x080a),***
- ***rsa_pss_pss with sha512(0x080b)***

] and no other algorithms;

].

FCS_TLSC_EXT.1.6 The TSF [***does not provide***] the ability to configure the list of supported ciphersuites as defined in FCS_TLSC_EXT.1.1.

FCS_TLSC_EXT.1.7 The TSF shall prohibit the use of the following extensions:

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

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

FCS_TLSC_EXT.1.9 The TSF shall [***reject [TLS 1.2, TLS 1.3] renegotiation attempts***].

FCS_TLSS_EXT.1 – TLS Server Protocol

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

[

- ***TLS_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***
- ***TLS_AES_128_GCM_SHA256***
- ***TLS_AES_256_GCM_SHA384***

] and no other ciphersuites.

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

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

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

].

Application Note: *For the management and GlobalProtect (GP) connections, the TOE is the TLS server.*

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

FCS_TLSS_EXT.1.5 The TSF [***provides***] the ability to configure the list of supported ciphersuites as defined in FCS_TLSS_EXT.1.1.

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

- Early data extension

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

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

FCS_IPSEC_EXT.1 – Ipsec Protocol

Application Note: *This SFR is defined in [NDcPP] but several elements were modified per guidance in [VPNGW-Module].*

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

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

FCS_IPSEC_EXT.1.3 The TSF shall implement [***tunnel mode***].

FCS_IPSEC_EXT.1.4 The TSF shall implement the IPsec protocol ESP as defined by RFC 4303 using the cryptographic algorithms **[*AES-CBC-128, AES-CBC-256 (specified in RFC 3602), AES-GCM-128, AES-GCM-256 (specified in RFC 4106)*] and [*AES-CBC-192 (specified in RFC 3602)*]** together with a Secure Hash Algorithm (SHA)-based HMAC [***HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512***].

FCS_IPSEC_EXT.1.5 The TSF shall implement the protocol: [

- *IKEv2 as defined in RFC 7296 and [**with mandatory support for NAT traversal as specified in RFC 7296, section 2.23**], and [**RFC 4868 for hash functions**]*

].

FCS_IPSEC_EXT.1.6 The TSF shall ensure the encrypted payload in the [***IKEv2***] protocol uses the cryptographic algorithms [***AES-CBC-128, AES-CBC-192, AES-CBC-256 (specified in RFC 3602), AES-GCM-128, AES-GCM-256 (specified in RFC 5282)***].

FCS_IPSEC_EXT.1.7 The TSF shall ensure that [

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

  [

  - ***length of time, where the time values can configured within [1 to 8760]* hours**

]].

FCS_IPSEC_EXT.1.8 The TSF shall ensure that [

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

  [

  - ***number of bytes;***
  - ***length of time, where the time values can be configured within [1 to 8760] hours;***

]].

FCS_IPSEC_EXT.1.9 The TSF shall generate the secret value x used in the IKE Diffie-Hellman key exchange ("x" in g^x mod p) using the random bit generator specified in FCS_RBG_EXT.1, and having a length of at least [**225 (for DH Group 14), 275 (for DH Group 15), 325 (for DH Group 16), 256 (for DH Group 19), 384 (for DH Group 20), 521 (for DH Group 21)**] bits.

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

[

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

] .

FCS_IPSEC_EXT.1.11 The TSF shall ensure that IKE protocols implement DH Group(s)

· 19 (256-bit Random ECP), 20 (384-bit Random ECP) according to RFC 5114 and [

· [14 (2048-bit MODP), 15 (3072-bit MODP), 16 (4096-bit MODP)] according to RFC 3526,

- ***[21 (521-bit Random ECP)] according to RFC 5114]***

].

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

FCS_IPSEC_EXT.1.13 The TSF shall ensure that [***IKEv2***] protocols perform peer authentication using a ***[RSA***] that use X.509v3 certificates that conform to RFC 4945 and [***Pre-shared Keys that conform to RFC 8784***].

FCS_IPSEC_EXT.1.14 The TSF shall only establish a trusted channel if the presented identifier in the received certificate matches the configured reference identifier, where the presented and reference identifiers are of the following fields and types: **Distinguished Name (DN)**, [***SAN: IP address, SAN: Fully Qualified Domain Name (FQDN)***].

Application Note: FCS_IPSEC_EXT.1.4, .1.11, and .1.14 are defined in the [NDcPP] but have been modified based on [VPNGW-Module].

5.2.3 User Data Protection (FDP)

FDP_RIP.2 – Full Residual Information Protection

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

5.2.4 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 remote Administrator from successfully establishing a 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:

a. Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters: [***"!", "@", "#", "$", "%", "^", "&", "*", "(", ")", [", "+", ",", "-", ".", "/", ":", ";", "<", "=", ">", "[", "\", "]", "_", "`", "{", "}", "~", and "all Unicode characters"***]];

b. Minimum password length shall be configurable to between [***8***] and [***16***] 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 Request/Response]***].

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_UIA_EXT.1.3 The TSF shall provide the following remote authentication mechanisms [***Web GUI password***, ***SSH password***, ***SSH public key***, ***X.509 certificate***] and local authentication mechanisms [***none***].

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

FIA_X509_EXT.1/Rev – X.509 Certificate Validation

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

- RFC 5280 certificate validation and certificate path validation **supporting a minimum path length of three certificates**.
- The certificate path must terminate with a trusted CA certificate designated as a trust anchor.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of the certificate using [**the Online Certificate Status Protocol (OCSP) as specified in RFC 6960, a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3, Certificate Revocation List (CRL) as specified in RFC 5759 Section 5**].
- The TSF shall validate the extendedKeyUsage field according to the following rules:
  - *Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.*
  - *Server certificates presented for DTLS/TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.*
  - *Client certificates presented for DTLS/TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.*
  - *OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.*

FIA_X509_EXT.1.2/Rev The TSF shall only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.

FIA_X509_EXT.2 – X.509 Certificate Authentication

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

Application Note: *This SFR is defined in [NDcPP] but was modified per guidance in [VPNGW-Module].*

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 to not accept the server certificate and fail the connection (not configurable).*

FIA_X509_EXT.3 – X.509 Certificate Requests

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

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

Application Note: *FIA_X509_EXT.3 is mandated by [VPNGW-Module].*

FIA_PSK_EXT.1 – Pre-Shared Key Composition

FIA_PSK_EXT.1.1 The TSF shall be able to use pre-shared keys for IPsec and [***IKEv2***].

FIA_PSK_EXT.1.2 The TSF shall be able to accept the following as pre-shared keys: [***generated bit-based***] keys.

FIA_PSK_EXT.2 – Generated Pre-Shared Keys

FIA_PSK_EXT.2.1 The TSF shall be able to [

- ***accept externally generated pre-shared keys***
- ***generate [128, 256] bit-based pre-shared keys via FCS_RBG_EXT.1.***

]

5.2.5 Security Management (FMT)

FMT_MOF.1/ManualUpdate – Management of Security Functions Behaviour

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

FMT_MOF.1/Services – Management of Security Functions Behaviour

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

FMT_MTD.1/CryptoKeys – Management of TSF Data

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

Application Notes: *This SFR is defined in [NDcPP] but was modified per guidance in [VPNGW-Module].*

FMT_MTD.1/CoreData – Management of TSF Data

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

FMT_SMF.1 – Specification of Management Functions

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

- *Ability to administer the TOE remotely;*
- *Ability to configure the access banner;*
- *Ability to configure the session inactivity time before session termination;*
- *Ability to update the TOE, and to verify the updates using* ***digital signature*** *capability prior to installing those updates;*

[

- ***Ability to manage the cryptographic keys;***
- ***Ability to configure the cryptographic functionality;***
- ***Ability to configure the list of supported (D)TLS ciphers;***
- ***Ability to configure the lifetime for IPsec Sas;***

- ***Ability to generate Certificate Signing Request (CSR) and process CA certificate response;***
- ***Ability to start and stop services***
- ***Ability to configure audit behaviour (e.g. changes to storage locations for audit; changes to behavior when local audit storage space is full);***
- ***Ability to modify the behaviour of the transmission of audit data to an external IT entity***
- ***Ability to 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 thresholds for SSH rekeying;***
- ***Ability to set the time which is used for time-stamps;***
- ***Ability to configure the reference identifier for the peer;***
- ***Ability to manage the TOE's trust store and designate X509v3 certificates as trust anchors;***
- ***Ability to manage the trusted public keys database;***
- ***Ability to configure the authentication failure parameters for FIA_AFL.1***

].

FMT_SMF.1/FFW – Specification of Management Functions

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

- *Ability to configure firewall rules;*

FMT_SMF.1/VPN – Specification of Management Functions

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

- ***Definition of packet filtering rules;***
- ***Association of packet filtering rules to network interfaces;***
- ***Ordering of packet filtering rules by priority; [***
- ***No other capabilities]*].**

FMT_SMF.1/IPS – Specification of Management Functions (IPS)

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

- *Enable, disable signatures applied to sensor interfaces, and determine the behavior of IPS functionality*
- *Modify these parameters that define the network traffic to be collected and analyzed:*
  - *Source IP addresses (host address and network address)*
  - *Destination IP addresses (host address and network address)*
  - *Source port (TCP and UDP)*
  - *Destination port (TCP and UDP)*
  - *Protocol (Ipv4 and Ipv6)*
  - *ICMP type and code*
- *Update (import) signatures*
- *Create custom signatures*
- *Configure anomaly detection*
- *Enable and disable actions to be taken when signature or anomaly matches are detected*
- *Modify thresholds that trigger IPS reactions*
- *Modify the duration of traffic blocking actions*
- *Modify the known-good and known-bad lists (of IP addresses or address ranges)*

- *Configure the known-good and known-bad lists to override signature-based IPS policies].*

FMT_SMR.2 – Restrictions on Security Roles

FMT_SMR.2.1	The TSF shall maintain the roles: ● *Security Administrator.*
FMT_SMR.2.2	The TSF shall be able to associate users with roles.
FMT_SMR.2.3	The TSF shall ensure that the conditions ● *The Security Administrator role shall be able to administer the TOE remotely;* are satisfied.

5.2.6 Protection of the TSF (FPT)

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

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

FPT_APW_EXT.1 – Protection of Administrator Passwords

FPT_APW_EXT.1.1	The TSF shall store administrative passwords in non-plaintext form.
FPT_APW_EXT.1.2	The TSF shall prevent the reading of plaintext administrative passwords.

FPT_FLS.1/SelfTest – Fail Secure (Self-test Failures)

FPT_FLS.1.1/SelfTest	The TSF shall **shut down** when the following types of failures occur: [*failure of the power-on self-tests, failure of integrity check of the TSF executable image, failure of noise source health tests*].

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***].

FPT_TST_EXT.1 – TSF Testing

FPT_TST_EXT.1.1	The TSF shall run a suite of the following self-tests:
	● *During initial start-up (on power on) to verify the integrity of the TOE firmware and software;*
	● *Prior to providing any cryptographic service and [**on-demand**] to verify correct operation of cryptographic implementation necessary to fulfil the TSF;*
	● [***start-up***] self-tests [
	  ● ***AES 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***
- ***SP 800-135 KDF Known Answer Tests***

].

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

Application Note: *Modified per TD0824 and TD0836.*

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

FPT_TST_EXT.3 – Self-Test with Defined Methods

FPT_TST_EXT.3.1 The TSF shall run a suite of the following self-tests [*[when loaded for execution]*] to demonstrate the correct operation of the TSF: [*integrity verification of stored executable code*].

FPT_TST_EXT.3.2 The TSF shall execute the self-testing through [*a TSF-provided cryptographic service specified in FCS_COP.1/**SigGen***].

FPT_TUD_EXT.1 – Trusted Update

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

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

FPT_TUD_EXT.1.3 The TSF shall provide a means to authenticate firmware/software updates to the TOE using a digital signature mechanism and *[**no other mechanisms***] prior to installing those updates.

Application Note: *Modified per TD0824.*

5.2.7 TOE Access (FTA)

FTA_SSL.3 – TSF-initiated Termination

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

FTA_SSL.4 – User-initiated Termination

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

FTA_TAB.1 – Default TOE Access Banners

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

5.2.8 Trusted Path/Channels (FTP)

FTP_ITC.1 – Inter-TSF Trusted Channel

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

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

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

- ***Transmitting audit records to an audit server using IPsec or TLS,***

].

Application Note: *The connection to the audit server (over TLS) can be tunneled over IPsec using the VPN connection identified below.*

FTP_ITC.1/VPN – Inter-TSF Trusted Channel (VPN Communications)

FTP_ITC.1.1/VPN The TSF shall **be capable of using IPsec to** provide a communication channel between itself and **authorized IT entities supporting VPN communications** 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/VPN The TSF shall permit [*the authorized IT entities*] to initiate communication via the trusted channel.

FTP_ITC.1.3/VPN The TSF shall initiate communication via the trusted channel for [***remote VPN gateways or peers***].

FTP_TRP.1/Admin – Trusted Path

FTP_TRP.1.1/Admin The TSF shall **be capable of using** [***SSH***, ***HTTPS***, ***IPsec***] **to** provide a communication path between itself and **authorized** remote **Administrators** ~~users~~ that is logically distinct from other communication paths and provides assured identification of its end points and

protection of the communicated data from disclosure **and provides detection of modification of the channel data**.

FTP_TRP.1.2/Admin The TSF shall permit remote **Administrators** ~~users~~ to initiate communication via the trusted path.

FTP_TRP.1.3/Admin The TSF shall require the use of the trusted path for *initial Administrator authentication and all remote administrative actions*.

5.2.9 Firewall (FFW)

FFW_RUL_EXT.1 – Stateful Traffic Filtering

FFW_RUL_EXT.1.1 The TSF shall perform stateful traffic filtering on network packets processed by the TOE.

FFW_RUL_EXT.1.2 The TSF shall allow the definition of stateful traffic filtering rules using the following network protocol fields:

- *ICMPv4*
  - *Type*
  - *Code*
- *ICMPv6*
  - *Type*
  - *Code*
- *IPv4*
  - *Source address*
  - *Destination Address*
  - *Transport Layer Protocol*
- *IPv6*
  - *Source address*
  - *Destination Address*
  - *Transport Layer Protocol*
  - ***[IPv6 Extension header type [Next Header, Hdr Ext Len, Header Specific Data, Option Type, Opt Data Len, Option Data, Routing Type]]***
- *TCP*
  - *Source Port*
  - *Destination Port*
- *UDP*
  - *Source Port*
  - *Destination Port.*

and distinct interface.

FFW_RUL_EXT.1.3 The TSF shall allow the following operations to be associated with stateful traffic filtering rules: permit or drop with the capability to log the operation.

FFW_RUL_EXT.1.4 The TSF shall allow the stateful traffic filtering rules to be assigned to each distinct network interface.

FFW_RUL_EXT.1.5 The TSF shall:

a) accept a network packet without further processing of stateful traffic filtering rules if it matches an allowed established session for the following protocols: TCP, UDP, [***ICMP***] based on the following network packet attributes:

1. *TCP: source and destination addresses, source and destination ports, sequence number, Flags;*
2. *UDP: source and destination addresses, source and destination ports;*
3. *[**ICMP: source and destination addresses, type, [code]**].*

b) Remove existing traffic flows from the set of established traffic flows based on the following: [***session inactivity timeout, completion of the expected information flow***].

FFW_RUL_EXT.1.6 The TSF shall enforce the following default stateful traffic filtering rules on all network traffic:

*a) The TSF shall drop and be capable of [**logging**] packets which are invalid fragments;*

*b) The TSF shall drop and be capable of [**logging**] fragmented packets which cannot be re-assembled completely;*

c) The TSF shall drop and be capable of logging packets where the source address of the network packet is defined as being on a broadcast network;

d) The TSF shall drop and be capable of logging packets where the source address of the network packet is defined as being on a multicast network;

e) The TSF shall drop and be capable of logging network packets where the source address of the network packet is defined as being a loopback address;

f) The TSF shall drop and be capable of logging network packets where the source or destination address of the network packet is defined as being unspecified (i.e. 0.0.0.0) or an address "reserved for future use" (i.e. 240.0.0.0/4) as specified in RFC 5735 for IPv4;

g) The TSF shall drop and be capable of logging network packets where the source or destination address of the network packet is defined as an "unspecified address" or an address "reserved for future definition and use" (i.e. unicast addresses not in this address range: 2000::/3) as specified in RFC 3513 for IPv6;

h) The TSF shall drop and be capable of logging network packets with the IP options: Loose Source Routing, Strict Source Routing, or Record Route specified; and

i) *[[*

- ***block both inbound and outbound IPv6 Site Local Unicast addresses (FEC0::/10)***
- ***block IPv6 Jumbo Payload datagrams (Option Type 194).***
- ***drop all inbound and outbound IPv6 packets containing a Hop-by-Hop header with option type values intended for Destination Options***
- ***block RFC 6598 "Carrier Grade NAT" IP address block of 100.64.0.0/10***
- ***drop all inbound IPv6 packets for which the layer 4 protocol and ports (undetermined transport) cannot be located.***
- ***drop all inbound IPv6 packets with a Type 0 Routing header***
- ***drop all inbound IPv6 packets with a Type 1 or Types 3 through 255 Routing Header.***
- ***drop all inbound IPv6 packets containing undefined header extensions/protocol values.***
- ***drop fragmented IPv6 packets when any fragment overlaps another.***
- ***drop all inbound IPv6 packets containing more than one Fragmentation Header within an IP header chain.***
- ***block IPv6 multicast addresses (FF00::/8) as a source address***

]].

FFW_RUL_EXT.1.7 The TSF shall be capable of dropping and logging according to the following rules:

a) *The TSF shall drop and be capable of logging network packets where the source address of the network packet is equal to the address of the network interface where the network packet was received;*

b) *The TSF shall drop and be capable of logging network packets where the source or destination address of the network packet is a link-local address;*

c) *The TSF shall drop and be capable of logging network packets where the source address of the network packet does not belong to the networks associated with the network interface where the network packet was received.*

FFW_RUL_EXT.1.8 The TSF shall process the applicable stateful traffic filtering rules in an administratively defined order.

FFW_RUL_EXT.1.9 The TSF shall deny packet flow if a matching rule is not identified.

FFW_RUL_EXT.1.10 The TSF shall be capable of limiting an administratively defined number of *half-open TCP connections. In the event that the configured limit is reached, new connection attempts shall be dropped and the drop event shall be [**counted, logged**].*

FFW_RUL_EXT.2 – Stateful Filtering of Dynamic Protocols

FFW_RUL_EXT.2.1 The TSF shall dynamically define rules or establish sessions allowing network traffic to flow for the following network protocols [***FTP***].

5.2.10 Packet Filtering (FPF)

FPF_RUL_EXT.1 – Rules for Packet Filtering

FPF_RUL_EXT.1.1 The TSF shall perform packet filtering on network packets processed by the TOE.

FPF_RUL_EXT.1.2 The TSF shall allow the definition of Packet Filtering rules using the following network protocols and protocol fields: [

- *IPv4 (RFC 791)*
  - *Source address*
  - *Destination Address*
  - *Protocol*
- *IPv6 (RFC 8200)*
  - *Source address*
  - *Destination Address*
  - *Next Header (Protocol)*
- *TCP (RFC 793)*
  - *Source Port*
  - *Destination Port*
- *UDP (RFC 768)*
  - *Source Port*
  - *Destination Port*].

FPF_RUL_EXT.1.3 The TSF shall allow the following operations to be associated with packet filtering rules: permit and drop with the capability to log the operation.

FPF_RUL_EXT.1.4 The TSF shall allow the packet filtering rules to be assigned to each distinct network interface.

FPF_RUL_EXT.1.5 The TSF shall process the applicable packet filtering rules (as determined in accordance with FPF_RUL_EXT.1.4) in the following order: [*Administrator-defined*].

FPF_RUL_EXT.1.6 The TSF shall drop traffic if a matching rule is not identified.

5.2.11 Intrusion Prevention (IPS)

IPS_ABD_EXT.1 – Anomaly-Based IPS Functionality

IPS_ABD_EXT.1.1 The TSF shall support the definition of [***baselines ('expected and approved')***] including the specification of [

- ***frequency***]

and the following network protocol fields:

- [***[IP addresses, ports, protocols, and data payloads]***]

IPS_ABD_EXT.1.2 The TSF shall support the definition of anomaly activity through [***manual configuration by administrators***].

IPS_ABD_EXT.1.3 The TSF shall allow the following operations to be associated with anomaly-based IPS policies:

- In any mode, for any sensor interface: [
  - ***allow the traffic flow***
  - ***send a TCP reset to the source address of the offending traffic***
  - ***send a TCP reset to the destination address of the offending traffic***]
- In inline mode:
  - [allow the traffic flow
  - block/drop the traffic flow
  - and [***no other actions***]].

IPS_IPB_EXT.1 – IP Blocking

IPS_IPB_EXT.1.1 The TSF shall support configuration and implementation of known-good and known-bad lists of [***source***] IP addresses and [***no additional address types***].

IPS_IPB_EXT.1.2 The TSF shall allow [Security Administrators] to configure the following IPS policy elements: [***known-good list rules, known-bad list rules, IP addresses***].

IPS_NTA_EXT.1 – Network Traffic Analysis

IPS_NTA_EXT.1.1 The TSF shall perform analysis of IP-based network traffic forwarded to the TOE's sensor interfaces, and detect violations of administratively-defined IPS policies.

IPS_NTA_EXT.1.2 The TSF shall process (be capable of inspecting) the following network traffic protocols:

- [Internet Protocol version 4 (IPv4), RFC 791
- Internet Protocol version 6 (IPv6), RFC 2460
- Internet control message protocol version 4 (ICMPv4), RFC 792
- Internet control message protocol version 6 (ICMPv6), RFC 2463
- Transmission Control Protocol (TCP), RFC 793
- User Data Protocol (UDP), RFC 768].

IPS_NTA_EXT.1.3 The TSF shall allow the signatures to be assigned to sensor interfaces configured for promiscuous mode, and to interfaces configured for inline mode, and support designation of one or more interfaces as 'management' for communication between the TOE and external entities without simultaneously being sensor interfaces.

- Promiscuous (listen-only) mode: [***physical network port or virtual-to-physical mapped interface port***];
- Inline (data pass-through) mode: [***physical network port or virtual-to-physical mapped interface port***];
- Management mode: [***physical MGMT port or virtual-to-physical mapped interface port***];
- [
  - ***no other interface types***]

IPS_SBD_EXT.1 – Signature-Based IPS Functionality

IPS_SBD_EXT.1.1 The TSF shall support inspection of packet header contents and be able to inspect at least the following header fields: [

- *IPv4: version; header length; packet length; ID; IP flags; fragment offset; time to live (TTL); protocol; header checksum; source address; destination address; IP options; and [**type of service (ToS)**].*
- *IPv6: version; payload length; next header; hop limit; source address; destination address; routing header; and [**traffic class, flow label**].*
- *ICMP: type; code; header checksum; and [**ID, sequence number**].*
- *TCP: source port; destination port; sequence number; acknowledgement number; offset; reserved; TCP flags; window; checksum; urgent pointer; and TCP options.*
- *UDP: source port; destination port; length; and UDP checksum*].

IPS_SBD_EXT.1.2 The TSF shall support inspection of packet payload data and be able to inspect at least the following data elements to perform string-based pattern-matching: [

- *ICMPv4 data: characters beyond the first 4 bytes of the ICMP header.*
- *ICMPv6 data: characters beyond the first 4 bytes of the ICMP header.*
- *TCP data (characters beyond the 20 byte TCP header), with support for detection of:*
  - *i. FTP (file transfer) commands: help, noop, stat, syst, user, abort, acct, allo, appe, cdup, cwd, dele, list, mkd, mode, nlst, pass, pasv, port, pass, quit, rein, rest, retr, rmd, rnfr, rnto, site, smnt, stor, stou, stru, and type.*
  - *ii. HTTP (web) commands and content: commands including GET and POST, and administrator- defined strings to match URLs/URIs, and web page content.*
  - *iii. SMTP (email) states: start state, SMTP commands state, mail header state, mail body state, abort state.*
  - *iv. [**no other types of TCP payload inspection**];*
- *UDP data: characters beyond the first 8 bytes of the UDP header;*

- *[**no other packet payload inspection**]*

IPS_SBD_EXT.1.3 The TSF shall be able to detect the following header-based signatures (using fields identified in IPS_SBD_EXT.1.1) at IPS sensor interfaces: [

a) IP Attacks
  i. IP Fragments Overlap (Teardrop attack, Bonk attack, or Boink attack)
  ii. IP source address equal to the IP destination (Land attack)
b) ICMP Attacks
  i. Fragmented ICMP Traffic (e.g. Nuke attack)
  ii. Large ICMP Traffic (Ping of Death attack)
c) TCP Attacks
  i. TCP NULL flags
  ii. TCP SYN+FIN flags
  iii. TCP FIN only flags
  iv. TCP SYN+RST flags
d) UDP Attacks
  i. UDP Bomb Attack
  ii. *UDP Chargen DDoS Attack*].

IPS_SBD_EXT.1.4 The TSF shall be able to detect all the following traffic-pattern detection signatures, and to have these signatures applied to IPS sensor interfaces: [

a) Flooding a host (DoS attack)
  i. ICMP flooding (Smurf attack, and ping flood)
  ii. TCP flooding (e.g. SYN flood)
b) Flooding a network (DoS attack)
c) Protocol and port scanning
  i. IP protocol scanning
  ii. TCP port scanning
  iii. UDP port scanning
  iv. ICMP scanning

IPS_SBD_EXT.1.5 The TSF shall allow the following operations to be associated with signature-based IPS policies:

- In any mode, for any sensor interface: [
  - ***allow the traffic flow;***
  - ***send a TCP reset to the source address of the offending traffic;***
  - ***send a TCP reset to the destination address of the offending traffic***]
- In inline mode:
  - block/drop the traffic flow;
  - and [
    - ***allow all traffic flow***
    - ***allow the traffic flow with following exceptions [some malicious traffic is always dropped]***]

IPS_SBD_EXT.1.6 The TSF shall support stream reassembly or equivalent to detect malicious payload even if it is split across multiple non-fragmented packets.

5.3 TOE Security Assurance Requirements

The security assurance requirements for the TOE are included by reference to [NDcPP], [FW-Module], [VPNGW-Module], and [IPS-Module].

Table 7 Assurance Components

Requirement Class	Requirement Component
ADV: Development	ADV_FSP.1: Basic functional specification
AGD: Guidance Documents	AGD_OPE.1: Operational user guidance
	AGD_PRE.1: Preparative procedures
ALC: Life-Cycle Support	ALC_CMC.1: Labelling of the TOE
	ALC_CMS.1: TOE CM coverage
	ALC_FLR.3: Systematic flaw remediation
ASE: Security Target Evaluation	ASE_INT.1: ST introduction
	ASE_CCL.1: Conformance claims
	ASE_SPD.1: Security problem definition
	ASE_OBJ.1: Security objectives for the operational environment
	ASE_ECD.1: Extended components definition
	ASE_REQ.1: Stated security requirements
	ASE_TSS.1: TOE summary specification
ATE: Tests	ATE_IND.1: Independent testing - conformance
AVA: Vulnerability Assessment	AVA_VAN.1: Vulnerability survey

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

- Supporting Document Mandatory Technical Document: Evaluation Activities for Network Device cPP, December-2019, Version 2.2
- Supporting Document Mandatory Technical Document: Evaluation Activities for Stateful Traffic Filter Firewalls PP-Module, June-2020, Version 1.4 + Errata 20200625
- Supporting Document Mandatory Technical Document: PP-Module for Virtual Private Network (VPN) Gateways Version 1.2, 2022-03-31
- Supporting Document Mandatory Technical Document PP-Module for Intrusion Prevention Systems (IPS) Version 1.0 2021-05-11

6. TOE Summary Specification

This chapter describes the security functions:

- Security Audit
- Cryptographic Support
- User Data Protection
- Identification and Authentication
- Security Management
- Protection of the TSF
- TOE Access
- Trusted Path/Channels
- Stateful Traffic Filtering
- Packet Filtering

6.1 Security Audit

FAU_GEN.1 FAU_GEN.1/VPN FAU_GEN.1/IPS	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, API, or CLI, as well as all of the events identified in Table 4 (which corresponds to the audit events specified in the [NDcPP]and [FW-Module]) and Tables 5 and 6 (which corresponds to the audit events specified in the [VPNGW-Module] and [IPS-Module]). The TOE supports "suppression" which is used to log a fixed number of similar threats over a period of time once with repeat counter to indicate total number of threats. Log suppression, when enabled, is a feature that instructs the TOE to combine multiple similar logs into a single log entry on the Traffic or Threat Monitor page. All log records include the following contents: date/time, event type, reaction (for threat logs), 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. Refer to the CC AGD [CCECG] for the complete list of mandated audit logs and contents.
FAU_GEN.2	The TOE identifies the responsible user for each event based on the specific username and/or network entity (identified by source IP address) that caused the event.
FAU_STG_EXT.1 FAU_STG.1	The audit trail generated by the standalone TOE comprises several logs, which are locally stored in the TOE file system on the hard disk and viewed via the CLI or Monitor pages: • 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. • Traffic logs—record the traffic flow events • Threat logs—record the detection and blocking of threats The size of each log file is administrator configurable by specifying the percentage of space allocated to each log type on the hard disk (minimum 45 MB). If the log size is reduced, the TOE removes the oldest logs when the changes are committed. When a log

	reaches the maximum size, the TOE starts overwriting the oldest log entries with the new log entries. Maximum disk space is platform dependent, and it depends on the hard disk drive installed on the system. By default, the TOE allocates 1-5% to system log, 1-5% to configuration log, 20-35% to traffic log, and 10-20% to threat log. For example, for a 120GB drive approximately 100GB is allocated for logging. Platform capabilities range from a limit of 3-4GB for the PA-410 which has a 16GB flash drive and up for the larger platforms (for example, PA-5220 has 1.70 TB drive). The standalone 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 TOE does not provide an interface where a user can modify the audit records, thus it prevents modification to the audit records. The standalone TOE can be configured to send generated audit records to an external Syslog server in real-time using TLS or IPsec. 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.1/IKE
FCS_CKM.2
FCS_COP.1/*
FCS_RBG_EXT.1

The TOE includes NIST-validated cryptographic algorithms provided by Palo Alto Networks Crypto Module supporting the cryptographic functions below. The following functions have been certified in accordance with the identified standards.

Table 8 Cryptographic Functions

Functions	Standards	Certificates
Asymmetric Key Generation and Asymmetric Key Generation/IKE (FCS_CKM.1)		
ECC key pair generation (NIST curves P-256, P-384, P-521) *Note that TLS and SSH each use any of P-256, P-384, or P-521 for authentication, but certificate generation by the TOE for P-256 and P-384 key sizes only.*	FIPS PUB 186-4	**Appliances (#A3453):** ECC RSA FFC **VMs (#A3454):** ECC RSA FFC
RSA key generation (key sizes 2048, 3072, 4096 bits)	FIPS PUB 186-4	
FFC Schemes using Diffie-Hellman groups that meet the following: RFC 3526 and SP 800-56Ar3 (2048-bit MODP, 3072-bit MODP, 4096-bit MODP)	RFC 3526 NIST SP 800-56Ar3	
Cryptographic Key Establishment (FCS_CKM.2)		

	ECDSA based key establishment (NIST P-256, P-384, P-521)	NIST SP 800-56A	**Appliances (#A3453):** ECC FFC **VMs (#A3454):** ECC FFC
	Key establishment scheme using Diffie-Hellman groups that meet the following: RFC 3526 and SP 800-56Ar3 (2048-bit MODP, 3072-bit MODP, 4096-bit MODP)	RFC 3526 NIST SP 800-56Ar3	
	AES Data Encryption/Decryption (FCS_COP.1/DataEncryption)		
	AES CBC, CTR, GCM (128, 192, 256 bits) *The 192 bits AES key size is used in CBC mode only and for IPsec only.*	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 (#A3453):** AES **VMs (#A3454):** AES
	Signature Generation and Verification (FCS_COP.1/SigGen)		
	RSA Digital Signature Algorithm (rDSA) (modulus 2048, 3072, 4096)	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 (#A3453):** RSA **VMs (#A3454):** RSA
	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 (#A3453):** ECDSA **VMs (#A3454):** ECDSA
	Cryptographic Hashing (FCS_COP.1/Hash)		
	SHA-1, SHA-256, SHA-384 and SHA-512 (digest sizes 160, 256, 384 and 512 bits)	ISO/IEC 10118-3:2004	**Appliances (#A3453):** SHS **VMs (#A3454):** SHS
	Keyed-hash Message Authentication (FCS_COP.1/KeyedHash)		

• 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 (#A3453):** HMAC **VMs (#A3454):** HMAC
Random Bit Generation (FCS_RBG_EXT.1)		
CTR_DRBG (AES-256)	ISO/IEC 18031:2011	**Appliances (#A3453):** DRBG **VMs (#A3454):** DRBG

The TOE implements the ISO/IEC 18031:2011 Deterministic Random Bit Generator (DRBG) based on the AES 256 block cipher in counter mode (CTR_DRBG(AES)). The TOE instantiates the DRBG with maximum security strength, obtaining the 256 bits of entropy to seed the DRBG. The hardware-based entropy source is described in the proprietary Entropy Design document. The TOE generates asymmetric cryptographic keys used for key establishment in accordance with FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3 for RSA schemes, FIPS PUB 186-4, "Digital Signature Standard (DSS)" and Appendix B.4 for ECC schemes, and NIST SP 800-56Ar3 and RFC 3526 for FFC schemes. In FIPS-CC mode, all non-Approved key generation and key establishment functions are disabled. Where required, only the CAVP-validated Approved key generation and key establishment functions are used for all protocols including TLS, HTTPS, IKE/IPsec, and SSH.

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 9 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.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 key establishment in accordance with NIST Special Publication 800-56A for elliptic curve-based key establishment[5] schemes, and NIST Special Publication 800-56A for finite field-based key establishment[6] schemes. The TOE acts as both a sender and as a recipient for all supported key establishment schemes (ECC, FFC, DH Group14). For TLS, the domain parameters used for the finite field-based key establishment scheme are compliance with FIPS 186-4. For SSH and IKE, the TOE uses safe primes as defined by RFC 3526[7] section 3 key establishment scheme. Per SP 800-56Ar3, only the safe primes (e.g., MODP-2048) defined in Sections 3 of RFC 3526 can be used. NOTE: RFC 7919 (FFDHE) is not supported.

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

FCS_CKM.1:
- X.509 key pair generation: ECDSA (256, 384), RSA (2048, 3072, 4096)
- SSH RSA key pair generation: RSA (2048, 3072, 4096)
- SSH: ECDSA (256, 384, 521)
- TLS: ECDSA (256, 384, 521), FFC (2048 or greater)
- IPsec: FFC DH Groups 14, 15, 16, ECDSA (256, 384, 521)

FCS_CKM.2:
- SSH: ECC (256, 384, 521)
- TLS: ECC (256, 384, 521), FFC (2048 or greater)
- IKE/IPsec: FFC DH Groups 14, 15, 16, ECC (256, 384, 521)[8]

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

FCS_CKM.4

Table 10 Private Keys and CSPs

CSP #	CSP/Key Name	Type	Description
1	RSA Private Keys	RSA	RSA Private keys for verification of signatures or authentication (RSA 2048, 3072, or 4096-bit)
2	ECDSA Private Keys	ECDSA	ECDSA Private key for verification of signatures and authentication (P-256, P-384, P-521)
3	TLS Pre-Master Secret	TLS Secret	Secret value used to derive the TLS session keys

[5] Elliptic curve-based key generation and establishment are used for HTTPS, TLS, SSH, and IKE connections.
[6] Finite field-based key generation and establishment are used for HTTPS and TLS connections.
[7] Diffie-Hellman Group 14 with safe primes based on RFC 3526 and SP 800-56Ar3 are used for SSH and IKE connections.
[8] Corresponding to Groups 19, 20, and 21.

	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-SHA2-256, HMAC-SHA2-384)
	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-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 and 256 bits in CBC and CTR, or 128 and 256 bits in GCM)
	9	SSH ECDH Private Components	ECDH	Diffie Hellman private component used in key establishment (ECDH 256, 384, 521)
	10	S-S VPN IPsec/IKE authentication Keys	HMAC	(SHA-256, SHA-384 or SHA-512) Used to authenticate the peer in an IKE/IPsec tunnel connection.
	11	S-S VPN IPsec/IKE session Keys	AES	Used to encrypt IKE/IPsec data. These are AES (128, 192, 256 CBC and 128, 256 GCM) IKE keys and (128, 192, and 256 CBC, 128 and 256 GCM) IPsec keys
	12	S-S VPN IPsec/IKE Diffie Hellman Private Components	DH	Diffie-Hellman (Group 14, 15, 16, 19, 20, and 21) private component used in key establishment
	13	RA VPN IPsec session Keys	AES	(128 CBC, 128 and 256 GCM) Used to encrypt remote access sessions utilizing IPsec.
	14	RA VPN IPsec authentication HMAC	HMAC	(SHA-256, SHA-384, SHA-512) Used in authentication of remote access IPsec data.

	15	Firmware code integrity check	HMAC ECDSA	Used to check the integrity of crypto-related code. (HMAC-SHA-256 and ECDSA P-256)
	16	Firmware Content Encryption Key	AES-256	Used to encrypt/decrypt firmware, software, and sensitive content.
	17	Password	Password	Authentication string with a minimum length of 8 characters. Stored hashed with SHA-256 and nonce.
	18	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 is the Firmware Content Encryption Key (also known as the Master Key). The KEK is not stored encrypted but is protected using Cryptod (Palo Alto Networks proprietary keys storage module) and destroyed by the TOE's overwriting method. The TOE zeroizes (i.e., overwrite) non-persistent cryptographic keys as soon as their associated session has terminated. In addition, the TOE recognizes when a private key expires and promptly zeroizes the key on expiration. The TOE does not permit expired private signature keys to be archived.

Private cryptographic keys, plaintext cryptographic keys, and all other critical security parameters stored in intermediate locations for the purposes of transferring the key/critical security parameters (CSPs) to another location are zeroized immediately following the transfer. Zeroization is done by overwriting the storage location with a random pattern, followed by a read-verify. Note that plaintext cryptographic keys and CSPs are only ever stored in volatile memory. For non-volatile memories other than EEPROM and Flash, the zeroization is executed by overwriting three times using a different alternating data pattern each time. This includes the SSD storage. This includes all CSPs that are not stored in volatile memory such as private keys, hashed passwords, and entropy seeds. The old KEK is overwritten when a new KEK is generated by an Approved DRBG.

For volatile memory and non-volatile EEPROM and Flash memories, the zeroization is executed by a single direct overwrite consisting of a pseudo random pattern generated by Approved DRBG. Sensitive data in volatile memory includes session keys such as encryption keys, integrity keys, pre-Master secret, etc.

FCS_HTTPS_EXT.1 FCS_TLSC_EXT.1 FCS_TLSS_EXT.1	The TOE can be configured as a TLS server for certificate-based authentication for secure connections (e.g., device, GP). The TOE uses TLS service profiles to specify a certificate and the allowed protocol versions for TLS services such as UI or TLS VPN. TLS PSK authentication is not supported by the TOE. The TOE (as a TLS client) uses TLS to initiate a TLS connection to external syslog server. The TOE (as TLS server) receives inbound remote administration TLS traffic on the management (MGT) interface from TLS client (e.g., web browser). The key agreement parameters of the server key exchange message consist of the key establishment parameters generated by the TOE: Diffie-Hellman Ephemeral parameters with key size of 2048, 3072, and 4096 bits MODP, ECDHE implementing NIST curves secp256r1, secp384r1, and secp521r1. The TOE supports session resumption using session tickets for a single context (no configuration needed). The TOE checks if session tickets expire which

	would trigger a full handshake. The session tickets are encrypted with AES encryption and 128-bits encryption key plus 256-bits HMAC-SHA-256 key; and adhered to the structural format provided in section 4 of RFC 5077. Fallback password-based authentication is not supported for this interface. The TOE denies connections from clients requesting connections using SSL 2.0, SSL 3.0, TLS 1.0, or TLS 1.1 and shall not establish a trusted channel if the fully qualified distinguished name (FQDN) in the subject or Subject Alternative Name (SAN) field contained in a certificate does not match the expected identifier for the peer. The TOE will match the FQDN identifier according to RFC 6125. The TOE also supports username as identifier for Web UI HTTPS connection. The TOE can be configured as a TLS server to permit inbound remote administration traffic (HTTPS) in which the client (e.g., web browser) initiates handshake and performs server authentication via server certificate with RSA key sizes 2048 bits or greater and ECDSA key sizes 256 bits or greater. The TOE's HTTPS protocol complies with RFC 2818 and is implemented using TLS 1.2 (RFC 5246) and TLS 1.3 (RFC 8446). HTTPS is used for Web UI management connection so only the server requirements in RFC 2818 are applicable. RFC 2818 (section 2.1: complies as specified, section 2.2: complies as specified, section 2.2.1: not applicable, section 2.2.2: complies as specified, section 2.3: default port is 443, section 2.4: Use 'https' as specified, section 3.1: not applicable, and section 3.2: client identity checking is performed). The key agreement parameters of the server key exchange message consist of the key establishment parameters generated by the TOE: Diffie-Hellman Ephemeral parameters with key size 2048, 3072, and 4096 bits MODP, ECDHE implementing NIST curves secp256r1, secp384r1, and secp521r1. The TOE denies connections from clients requesting connections using SSL 2.0, SSL 3.0, TLS 1.0 or TLS 1.1 by default (when FIPS-CC mode is enabled). The TOE can be configured as a TLS client for secure communication to an external audit server. The TOE presents the Supported Elliptic Curves/Supported Groups 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 (no override mechanism). The TOE compares the external server's presented identifier to the reference identifier by matching the certificate FQDN (hostname) of the server certificate. The SAN is checked first and if there is any match, the connection is allowed. The TOE supports wildcards (for FQDN only) for peer authentication. Certificate pinning is not supported. The TOE implements TLS 1.2 (RFC 5246) and TLS 1.3 (RFC 8446). TOE (as TLS client) to syslog server. Support TLSv1.2 and TLSv1.3, DHE (finite-field based for TLSv1.2 only), and ECDHE (elliptic curve-based) schemes. When FIPS-CC mode is enabled, only the following TLS ciphersuites are configured: ● 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_AES_128_GCM_SHA256 • TLS_AES_256_GCM_SHA384 TOE (as TLS server) for the Web UI management and GlobalProtect (GP) connections. Supports TLSv1.2 and TLSv1.3, and DHE (finite-field based for TLSv1.2 only), and ECDHE (elliptic curve-based) schemes. When FIPS-CC mode is enabled, only the following TLS ciphersuites are configured: • 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 • TLS_AES_128_GCM_SHA256 • TLS_AES_256_GCM_SHA384 When FIPS-CC mode is enabled, all TLS connections will only support the following signature algorithms extensions: • rsa_pkcs1 with sha256(0x0401), • rsa_pkcs1with sha384(0x0501), • rsa_pkcs1 with sha512(0x0601), • ecdsa_secp256r1 with sha256(0x0403), • ecdsa_secp384r1 with sha384(0x0503), • ecdsa_secp521r1 with sha512(0x0603), • rsa_pss_rsae with sha256(0x0804), • rsa_pss_rsae with sha384(0x0805), • rsa_pss_rsae with sha512(0x0806), • rsa_pss_pss with sha256(0x0809), • rsa_pss_pss with sha384(0x080a), • rsa_pss_pss with sha512(0x080b)
FCS_SSH_EXT.1 FCS_SSHS_EXT.1	The TOE supports SSHv2 server (compliant to RFCs 4251, 4252, 4253, 4254, 4256, 4344, 5647, 5656, 6668, 8308, 8332) with AES encryption/decryption algorithm (in CBC, CTR, or GCM mode [gcm@openssh.com]) with key sizes of 128 and 256 bits. No

	optional characteristics are supported. The TOE also supports HMAC-SHA-256, HMAC-SHA-512, implicit MAC (aes128-gcm@openssh.com and aes256-gcm@openssh.com) for integrity and authenticity. Both encryption and integrity algorithms are administrator-configurable but non-Approved algorithms such as 3DES, HMAC-MD5, diffie-hellman-group-1 are all disabled when FIPS-CC mode is enabled. Only the Approved encryption and integrity algorithms along with key exchange algorithms ecdh-sha2-nistp256, ecdh-sha2-nistp384, and ecdh-sha2-nistp521 (based on elliptic curve from FCS_CKM.2) and authentication public-key algorithms ssh-rsa, rsa-sha2-256, and rsa-sha-512 are permitted in the evaluated configuration. In addition, the TOE supports SSH KDF as specified in RFC 5656 section 4. If the SSH client (in the operational environment) only support non-Approved algorithms, the SSH connection will be rejected by the TOE. SSH server supports both RSA and ECDSA SSH host keys as specified in FCS_SSHS_EXT.1. The TOE uses OpenSSH implementation to support the SSHv2 connections. The password authentication timeout period is 60 seconds allowing clients to retry only 4 times. In addition, public-key (RSA), keyboard-interactive, and password-based authentication can be configured with password-based being the default method. For password, the TOE verifies the user identity when the username is entered. For public-key, the administrator must associate the public key to the user. Keyboard-interactive is a generic authentication method that can be used to implement different types of authentication mechanisms such as password and public-key. SSH packets are limited to 262,105 bytes and any packet over that size will be dropped (i.e., not processed farther and buffer containing the packet will be freed). The TOE manages a tracking mechanism for each SSH session so that it can initiate a new key exchange (rekey) when either a 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 and the threshold limits apply to both transmitted and received data.
FCS_IPSEC_EXT.1 FCS_CKM.1/IKE FIA_PSK_EXT.1 FIA_PSK_EXT.2	The TOE provides mechanisms to implement an IPsec Security Policy Database (SPD) in tunnel mode and to process packets to satisfy the behavior of DISCARD, BYPASS and PROTECT packet processing as described in RFC 4301. This is achieved through the administrator configuring appropriately specified access control lists (ACLs). The ACLs consist of policy rules and profiles. The TOE compares packets in turn against each rule in the Security ACL to determine if the packet matches the rule. Packets can be matched based on protocol (e.g., TCP, UDP), source IP address and destination IP address. The first rule that matches the traffic is applied. If a policy rule matching the traffic attributes is not found, or if it is found and it specifies a deny action, then the packet is dropped (or DISCARDed) and the session is deleted. If the application flow is allowed and no further security profiles are applied then it is forwarded (it is allowed to BYPASS the tunnel). If the application is allowed and there are additional security profiles set, it will be sent to the stream signature processor. The traffic matching the IPsec crypto Security profile would then flow through the IPsec tunnel and be classified as “PROTECTED”. If there is no SA that the IPsec can use to protect this traffic to the peer, IPsec uses IKE to negotiate with the remote peer to set up the necessary IPsec SAs on behalf of the data flow. The negotiation uses information specified in the IKE Network Profiles. If the TOE receives a packet that does not match any rules in the SPD the TOE discards the packet. By default, the TOE is configured to allow all intrazone (within the zone) traffic and deny all interzone (between zones) traffic. Typically, intrazone traffic is considered to be trusted; however, both intrazone and interzone traffic can be configured to deny all traffic if there is no rule match by clicking on the security policy and clicking on the Override button on the bottom on the **Policy > Security** screen. In the evaluated configuration, the default deny all rule for interzone traffic should not be modified.

	The TOE uses route-based VPNs where the firewall makes a routing decision based on the destination IP address. It is not necessary to define special rules or to make explicit reference to a VPN tunnel; routing and encryption decisions are determined only by the destination IP address. Packets matching the destination IP address are permitted otherwise they are denied. The TOE also supports Network Address Translation (NAT) policies (as defined in RFC 5996) where policies can be defined to specify whether source or destination IP addresses and ports are converted between public and private addresses and ports. For example, private source addresses can be translated to public addresses on traffic sent from an internal (trusted) zone to a public (untrusted) zone. NAT policy rules are based on the source and destination zones, the source and destination addresses, and the application service. The NAT policy rules are compared against the incoming traffic in sequence; the first rule that matches the incoming traffic is applied. If no rules match, then the flow is denied. IKEv2 SA lifetime limits can be configured by an authorized administrator and can be limited to 24 hours for phase 1 (range: 1 to 8760 hours) and 8 hours (range: 1 to 8760 hours) for phase 2 SAs. IKEv2 SA phase 2 lifetime limits can be established based on number of bytes. The IKEv2 (as defined in RFCs 5996 and 4868) protocols implemented by the TOE include DH Group 14 (2048-bit MODP), DH Group 15 (3072-bit MODP), DH Group 16 (4096-bit MODP), DH Groups 19 (256-bit Random ECP), 20 (384-bit Random ECP), and 21 (521-bit Random ECP) using RSA peer X.509v3 certificate authentication conforming to RFC4945. The DH Groups 14, 15, and 16 are based on RFC 3526 (sections 3, 4, and 5) and ECP is based on elliptic curve in FCS_CKM.2. IKE keys used as part of the establishment are generated according to FIPS 186-4. When the TOE initiates IKE negotiation, the DH group is sent in order according to the peer's configuration. When the TOE receives an IKE proposal, it will select the first match and the negotiation will fail if there is no match. In addition, the TSF will only establish an IKE channel if the presented identifier in the peer X.509v3 certificate CN (Common Name) or SAN (Subject Alternative Name) field matches the configured identifier: Distinguished Name (DN), IP address (in SAN), or Fully Qualified Domain Name (FQDN) in SAN. The keys are generated using the AES-CTR Deterministic Random Bit Generator (DRBG), as specified in ISO/IEC 18031:2011, and the following corresponding key sizes (in bits) are used: 256 (for DH Group 14), 384 (for DH Group 15), 384 (for DH Group 16), 256 (for DH Group 19), 384 (for DH Group 20), and 521 (for DH Group 21) bits. The nonces used in the IKE exchanges are generated at least 128 bits in size and at least half the output size of the PRF hash. The TSF generates the nonces of length 256 (for DH Groups 14, 15, 16, 19, 20, and 21) bits, and are generated with the Approved DRBG. The TSF generates the secret value x used in the IKE Diffie-Hellman key exchange ("x" in g^x mod p) using the random bit generator specified in FCS_RBG_EXT.1 and having a length of 256 (for DH Group 14), 384 (for DH Group 15), 384 (for DH Group 16), 256 (for DH Group 19), 384 (for DH Group 20), and 521 (for DH Group 21) bits. The TOE provides AES-CBC-128, AES-CBC-192, AES-CBC-256, AES-GCM-128, and AES-GCM-256 for encrypting IKEv2 payloads. The TOE provides AES-CBC-128, AES-CBC-192, AES-CBC-256, AES-GCM-128 and AES-GCM-256 for encrypting IPsec payloads. The administrator is instructed to ensure that the size of key used for ESP must be less than or equal to the key size used to protect the IKE payload. Once configured, the TSF will check that the symmetric key size used in the IKE is greater than or equal to the symmetric key size used in the ESP. The TOE implementation of IPsec protocol ESP complies with RFC 4303 using the algorithms specified in

	FCS_IPSEC_EXT.1.4 together with the HMAC specified in FCS_IPSEC_EXT.1.4. SHA-based HMAC function truncated output is not used. The TOE supports PPK (Post-Quantum Pre-shared Key) as specified in RFC 8784 for authentication with the regular IKEv2 key exchange. The TOE will generate PPK keys sizes of 128 bits (default) and 256 bits upon request. The bit-based keys are generated from the TOE's Approved DRBG in FIPS-CC mode, and keys must be securely transported via an out of band mechanism. Keys that are externally generated can be accepted and used by the TOE but the quality of those keys are out of scope.

6.3 User Data Protection

FDP_RIP.2	The TOE allocates and releases (i.e., deallocates) the memory resources used for network packet objects. When it receives new data from the network and allocates new buffer resources to store and transmit data to the network, it ensures that the new buffers are not padded out with previously transmitted or otherwise residual information by overwriting unused parts of the buffer with 0s.

6.4 Identification and Authentication

FIA_UIA_EXT.1	The TOE is designed to require users to be identified and authenticated before they can access any of the TOE functions. The only capabilities allowed prior to users authenticating are the display of the informative (login) banner and responding to ICMP request (e.g., ping or ICMP echo reply). The TOE maintains user accounts which it uses to control access to the TOE. When creating a new user account, the administrator specifies a username (i.e., user identity or ID), a password or X.509v3 certificate[9]/public-key/common access card, and a role. The TOE uses the username and password (i.e., API key for API) attributes to identify and authenticate the user when the user logs in via the GUI, API, 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. API authentication supports API key which is the password encrypted. 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, API, or CLI. The administrator can logon to the GUI/API by using a secure connection (HTTPS, HTTPS over IPsec) from a web browser or to CLI by using a secure connection (SSHv2) from a SSH client. The TOE provides remote access to the GUI/API/CLI using HTTPS/TLS/IPsec or SSHv2. 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

[9] X.509 certificate is only supported for the UI. It is not supported for SSH or API.

	TOE (e.g., verify the possession of a private key that corresponds to the public key defined for that user account).
FIA_PMG_EXT.1	Passwords can be composed of upper and lower case letters, numbers and special characters (“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(”, “)”, “‘”, “+”, “,”, “-”, “.”, “/”, “:”, “;”, “<”, “=”, “>”, “[”, “\”, “]”, “_”, “`”, “{”, “}”, “~”, and “all Unicode characters”). For any Admin, the password must be composed of at least one lower case, one upper case, and one number or special character. The minimum password length is configurable by the administrator from 8 up to 16 characters. Note in FIPS-CC mode, the minimum password length cannot be configured below 8. The maximum password length is 63 characters. For example, if the administrator configures the minimum password length as 15, users can only create passwords from length of 15 to 63.
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[10]) during which providing the correct credentials will still not allow access (i.e., locked out). 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 required 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. 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), IKE/IPsec, and HTTPS connections. Public key infrastructure (PKI) credentials, such as RSA keys and certificates are stored in the TOE’s underlying file system on the appliance. Certificates and their associated private key are stored in a single container: the Certificate File. The PKCS#12 file consists of an Encrypted Private Key and X.509 Certificate. By default, all the private keys are protected since they are always stored in encrypted format using AES-256. The physical security of the appliance (A.PHYSICAL_PROTECTION) protects the appliance and the certificates from being tampered with or deleted. In addition, the TOE identification and authentication security functions protect an unauthorized user from gaining access to the TOE. Revocation checking is performed on the TLS and IPsec peer certificates presented to the TOE. For IPsec, the TOE supports both Online Certificate Status Protocol (OCSP) and Certificate Revocation List (CRL) status verification for certificate profiles. If both are configured, the TOE first tries the OCSP method; if the OCSP server is unavailable, the TOE uses the CRL method. For TLS, the TOE only supports OCSP. The TOE uses the following rules for validating the extendedKeyUsage[11] field: ● Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field. ● OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field. The TOE validates a certificate path by ensuring the presence of the basicConstraints extension is present and the cA flag is set to TRUE for all CA certificates. The TOE

[10] The value of 0 is not allowed in the evaluated configuration.
[11] Certificates are not used for trusted updates or executable code integrity.

	forms a Certificate trust path by ensuring that the basic constraints are met, proper key usage parameters exist, the CA flag exists, performing a revocation check of each certificate in the path and performing the validity of the CA certificate. The TOE will not treat a certificate as a CA certificate if the basicConstraints extension is not present or the cA flag is not set to TRUE. The TOE supports certificate path validation for a minimum path length of three certificates and terminates with a trusted CA certificate (i.e., Root certificate). The TOE downloads and caches OCSP status information for every CA listed in the trusted CA list of the TOE. The OCSP status is cached for the 'next update time' that is configured on the OCSP responder. The TOE uses this received value as the cache time. OCSP responders can also be configured for other external devices if someone decides to use it. The TOE uses a hard coded 1 hour as next update time (cached time) in this case. Caching only applies to validated certificates; if a TOE never validated a certificate, the TOE cache does not store the OCSP information for the issuing CA. To use OCSP for verifying the revocation status of certificates, administrator must configure the TOE to access an OCSP responder (server). The entity that manages the OCSP responder can be a third-party certificate authority (CA) or, if the enterprise has its own PKI, the TOE itself. The TOE downloads and caches the last-issued CRL for every CA listed in the trusted CA list of the TOE. The signature on the CRL is verified as defined in RFCs 5280 and 5759 (supporting strong Suite B ECDSA algorithms and curve sizes for signing CRL). 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 behavior for syslog connection and cannot be changed. For the HTTPS management session and IKE/IPsec connection to another firewall (VPN gateway), this behavior is configurable but in the evaluated configuration, the required action is to block the TLS session. This is not done by default. To configure this option, create a Certificate Profile and check the "Block session if certificate status cannot be retrieved within timeout" checkbox.
FIA_X509_EXT.3	The authorized administrator may generate a certificate request as specified in RFC 2986 and provide the following information in the request: public key, Common Name, Organization, Organizational Unit, and Country. The administrator may also import a certificate and private key into the TOE from an enterprise certificate authority or obtain a certificate from an external CA. The TOE provides the ability for administrators to generate a Certificate Signing Request (CSR) with a multi-level organizational unit. When the administrators import a certificate based on the CSR, the TOE will validate the certificate chain is based on a trusted CA which must be present on the TOE. Otherwise, the TOE will reject the certificate and will not associate it with the CSR.

6.5 Security Management

FMT_MOF.1/ManualUpdate FMT_MOF.1/Services FMT_MTD.1/CoreData FMT_MTD.1/CryptoKeys	The TOE provides a GUI or API management interface, and CLI to support security management of the TOE. The GUI or API is accessible via secure connection to the management port remotely over HTTPS or HTTPS tunneled over IPsec. The CLI is accessible via secure connection to the management port on the device 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 examples, the ability to manage the TOE's trust store, enable or disable of start/stop services (e.g., DNS via Device > Setup > Services > DNS Servers, SNMP via Device > Setup > Operations > SNMP Setup), configure audit behavior, and VPN keys/certificates are restricted to Security Administrators only. The Security Administrators can generate RSA and ECDSA keypair (public key embedded in certificate), import private key and certificate, and delete keys. The key management operations can be performed via the Web UI or by CLI/API command. 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 FMT_SMF.1/FFW FMT_SMF.1/VPN FMT_SMF.1/IPS	The security management functions provided by the TOE include, but are not limited to: ● Ability to administer the TOE remotely; ● Ability to configure the access banner; ● Ability to configure the session inactivity time before session termination or locking; ● Ability to update the TOE, and to verify the updates using digital signature capability prior to installing those updates; ● Ability to configure the authentication failure parameters for FIA_AFL.1; ● Ability to configure the list of TOE-provided services available before an entity is identified and authenticated, as specified in FIA_UIA_EXT.1; ● Ability to configure the cryptographic functionality; ● Ability to configure thresholds for SSH rekeying; ● Ability to 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; ● Ability to configure firewall rules; ● Ability to configure the lifetime for IPsec SAs; ● Ability to configure the reference identifier for the peer; ● Ability to enable, disable, determine and modify the behavior of all the security functions of the TOE identified in the VPNGW Module to the Administrator; ● Ability to configure all security management functions identified in other sections of the VPNGW Module; ● Ability to configure the IPsec functionality; ● Ability to import X.509v3 certificates; ● Ability to configure audit behavior (e.g. changes to storage locations for audit; changes to behaviour when local audit storage space is full); ● Ability to modify the behaviour of the transmission of audit data to an external IT entity ● Ability to start and stop services;

	● Ability to manage the trusted public keys database; ● Definition of packet filtering rules; ● Association of packet filtering rules to network interfaces; ● Ordering of packet filtering rules by priority; ● Enable, disable signatures applied to sensor interfaces, and determine the behavior of IPS functionality; ● Modify these parameters that define the network traffic to be collected and analyzed: o Source IP addresses (host address and network address) o Destination IP addresses (host address and network address) o Source port (TCP and UDP) o Destination port (TCP and UDP) o Protocol (IPv4 and IPv6) o ICMP Type and Code ● Update (import) signatures; ● Create custom signatures; ● Configure anomaly detection; ● Enable and disable actions to be taken when signature or anomaly matches are detected; ● Modify thresholds that trigger IPS reactions; ● Modify the duration of traffic blocking actions; ● Modify the known-good and known-bad lists (of IP addresses or address ranges); ● Configure the known-good and known-bad lists to override signature-based IPS policies. The GUI, CLI, and API (XML and REST) provide the same supported management functionality. All management functions above are available from any of these interfaces.
FMT_SMR.2	The TOE controls user access to commands and resources based on user role. Users are given permission to access a set of commands and resources based on their user role. By default, the TOE has the following pre-defined administrator roles: Superuser, Superuser (Read-Only), and Device 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 (including Security Administrator as defined in the NDcPP) can administer the TOE remotely via CLI or Web UI, and a user account can only be assigned one role at a time. ● **Superuser**—Full read-write access to TOE and all device groups, templates, and managed firewalls including user and role management (create, modify, delete). ● **Superuser (Read-Only)**—Read-only access to TOE and all device groups, templates, and managed firewalls. ● **Device Administrator**—Full access to TOE except for the create,

	modify, or delete administrators or roles. The Security Administrator role shall be able to administer the TOE remotely.

6.6 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 X.509 Certificate. By default, all the private keys are protected since they are always stored in encrypted format using AES-256. The TOE prevents the reading of all keys by encrypting them with a Master Key using AES-256. The TOE does not provide an interface to read the Master Key. The TOE is designed specifically to prevent access to locally-stored cryptographically protected passwords and does not disclose any keys stored in the TOE. The TOE protects the confidentiality of user passwords by hashing the passwords using SHA-256. The TOE does not offer any functions that will disclose to any users a stored cryptographic key or password.
FPT_FLS.1/SelfTest FPT_TST_EXT.1 FPT_TST_EXT.3	The TOE meets FPT_TST_EXT requirements and therefore provides self-tests at start-up to demonstrate the correct operation of: key error detection, cryptographic algorithms, and DRBG. The self-tests also verify the integrity of stored TSF executable code and TSF data. The TOE performs the following Power-on self-tests: • AES 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 • SP 800-135 KDF Known Answer Tests A known-answer test involves operating the cryptographic algorithm on data for which the correct output is already known and comparing the calculated output with the previously generated output (the known answer). If the calculated output does not equal the known answer, the known-answer test shall fail. The firmware integrity test verifies the signature of the image to ensure that it has not been modified or tampered with it. If the signature verification fails, the integrity test will fail. The entropy health

	tests consist of two tests, RCT (Repetition Count Test) and APT (Adaptive Proportion Test). RCT is designed to detect if the noise source become stuck outputting the same value too many times in a row. APT is designed to detect if the frequency of a sample value occurs too many times in a particular window size. The entropy health test will fail if any of the two tests fails. If a self-test (e.g., known answer test, entropy health tests, firmware integrity test using ECDSA as defined in FCS_COP.1/SigGen) fails, the TOE enters an error state and outputs an error indicator. The TOE doesn't perform any cryptographic operations or functions while in the error state. All data output from the TOE is inhibited when an error state exists. Should one or more power-up self-tests fail the module will reboot and enter maintenance mode (i.e., error state) in which the reason for the failure can be determined. In the maintenance mode, all operational and network functions will be unavailable with one exception which is the Reboot operation. These self-tests are sufficient to ensure correct functionality of the TSF because the self-tests encompass the cryptographic functionality and the integrity of the entire TOE software/firmware executable code.
FPT_TUD_EXT.1	Authorized administrators may query the current software/firmware version of the TOE. • CLI: command 'show system info | match sw-version', • UI: Dashboard > General Information, • API: https://<TOE>/api/?type=op&cmd=<show><system><info></info></system></show>&key=<APIkey> 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) all content that is downloaded to the TOE. If the TOE is not connected to the internet, the administrators can download the updates and upload it to the TOE. When the TOE update package and its corresponding digital signature is downloaded or uploaded; the digital signature is checked automatically by TOE by verifying the signature using the public key (corresponding to the RSA key used to create the signature). Palo Alto Networks manages the updates.paloaltonetworks.com and guarantees that images are digitally signed. Public keys are stored and protected on the TOE's file system. If the signature is verified, the update is performed; otherwise, the update is not performed.
FPT_STM_EXT.1	The TOE is a hardware appliance, or a virtual appliance image installed on a virtualization platform that includes a hardware-based real-time clock. The clock is based on a quartz crystal running at high frequency and accuracy. The hardware hosting the VM-Series provides the time clock, as well as CPU, ports, etc., which are provided by VM hypervisor. The TOE's embedded OS manages the clock and exposes administrator clock-related functions such as set time. The clock is used for audit record time stamps, measuring session activity for termination, time-based lockout following authentication failure, SSH/IPsec session rekeying, and X.509 certificate validation.

6.7 TOE Access

FTA_SSL.3	The TOE subsequently will enforce an administrator-defined inactivity timeout value after which the remote session will be terminated regardless of authentication methods (e.g., password, public-key, X.509v3 certificate). The TOE can be configured by an administrator to set an interactive session timeout value (any integer value from 1 to 1,440 minutes with default set to 60 minutes). The function is enabled by default and the administrator must follow the CC AGD to configure the session idle timeout value. A remote session that is inactive (i.e., no commands issuing from the remote client) for the defined timeout value will be terminated. The users will be required to re-enter their user ID and their password or perform public-key or certificate-based authentication, so they can establish a new session once a session is terminated. Users can also initiate termination of their own sessions. They can click on the "Logout" link or enter the "exit" command.
FTA_SSL.4	The TOE provides remote administrators the ability to logout (or terminate) their sessions as directed by the user.
FTA_TAB.1	The TOE can be configured to display an informative banner that will appear prior to authentication when accessing the TOE via remote connection to the management port in order to access the Web Interface (HTTPS) or CLI (SSH).

6.8 Trusted Path/Channels

FTP_ITC.1 FTP_ITC.1/VPN	The TOE can be configured to send audit records to external syslog server(s) using TLS or IPsec in real-time. Tunneling TLS connection over IPsec can be done using a service route for additional security. The TOE permits the TSF to initiate communication with the syslog server using TLS trusted channel. The TOE permits the following communications below: • Connecting with remote VPN gateways/peers using IPsec [TOE is peer], • Connecting with GlobalProtect (VPN peer) using TLS [TOE is server], • Transmitting audit records to an audit server using IPsec or TLS [TOE is client], The TOE communicates with its authorized trusted entities over TLS or IPsec, and all communication are sent over the trusted channel and are protected by the security protocols. The underlying cryptographic algorithms and implementation are CAVP-validated and are part of the TOE.
FTP_TRP.1/Admin	The TOE provides SSH, HTTPS (TLSv1.3 and TLSv1.2), and IPsec[12] to support secure remote administration. Administrators can initiate a remote session that is secured (from disclosure and modification) using CAVP-validated cryptographic operations, and all remote security management functions require the use of a secure channel. In FIPS-CC mode, telnet and HTTP are permanently disabled.

6.9 Stateful Traffic Filtering

FFW_RUL_EXT.1 FFW_RUL_EXT.2	An authorized administrator may configure the TOE to apply stateful traffic filtering rules with actions permit with logging or drop with logging on the following protocols:

[12] The HTTPS management session can be tunneled over IPsec to provide additional security.

	● Internet Control Message Protocol version 4 (ICMPv4)
	● Internet Control Message Protocol version 6 (ICMPv6)
	● Internet Protocol version 4 (IPv4)
	● Internet Protocol version 6 (IPv6)
	  o IPv6 Extension header type (Next Header, Hdr Ext Len, Header Specific Data, Option Type, Opt Data Len, Option Data, Routing Type)
	● Transmission Control Protocol (TCP)
	● User Datagram Protocol (UDP)
	Conformance with the RFC 792 (ICMPv4), RFC 4443 (ICMPv6), RFC 791(IPv4), RFC 2460 (IPv6), RFC 793 (TCP), RFC 768 (UDP) protocols is verified by Palo Alto through regular quality assurance, regression, and interoperability testing.
	A Security Administrator can configure the TOE to control the type of information that is allowed to pass through the TOE. The Security Administrator defines the security zone and applies security policies and security profiles to network traffic attempting to traverse the TOE to determine what actions to take.
	Security Zones
	The TOE groups interfaces into security zones. Each zone identifies one or more interfaces on the TOE. Separate zones must be created for each type of interface (Layer 2, Layer 3, or virtual wire), and each interface must be assigned to a zone before it can process traffic.
	Security Policies
	Security policies provide the firewall rule sets that specify whether to block or allow network connections, based on the source and destination zones, addresses, and the application service (such as UDP port 67 or TCP port 80). Security policy rules are processed in sequence order (top to bottom), applying the first rule that matches the incoming traffic. To make a rule get checked first, the Security Administrator can place the rule at the top of the order. Conflicting rules can be configured but the first matching rule will be enforced on the traffic.
	Security policies can be defined only between zones of the same type and be assigned to a distinct network interface. However, the administrator can create a VLAN interface for one or more VLANs and then apply a security policy between the VLAN interface zone and a Layer 3 interface zone. This has the same effect as applying policies between the Layer 2 and Layer 3 interface zones.
	Each rule can be configured to generate a log record when the traffic matches the defined rule using the 'policy->Security->options' selection. The logging option can be configured to log at the start of a session, or at the end of a session or both.
	The TOE enforces the stateful traffic filtering rules based on the following subject and information security attributes:
	● Source security zone to which the physical network interface is assigned
	● Destination security zone to which the network interface is assigned
	● Information specifiable in security policies, which provide the information flow rule sets:
	  o presumed identity of source subject—source address information within the packet

	○ identity of destination subject—destination address information within the packet ○ transport layer protocol (e.g., TCP, UDP) ○ Internet layer protocol (e.g., ICMP type, code) ○ source subject service identifier (e.g., source port number) ○ destination subject service identifier (e.g., destination port number) ● Information security attributes for stateful packet inspection—for connection-oriented protocols (e.g., TCP), the sequence number, acknowledgement number, and flags (SYN, ACK, RST, FIN); and for connectionless protocols (e.g., UDP), the source and destination network identifiers; and source and destination service identifiers. Note that the TOE uses an IP-based network stack. The TOE supports the Transmission Control Protocol (TCP) (RFC 793) which performs a handshake during session setup to initiate and acknowledge a session. After the data is transferred, the session is closed in an orderly manner, where each side transmits a FIN packet and acknowledges it with an ACK packet. The handshake that initiates the TCP session is often a three-way handshake (an exchange of three messages) between the initiator and the listener, or it could be a variation, such as a four-way or five-way split handshake or a simultaneous open. The TOE supports the TCP Split Handshake Drop feature, which can prevent TCP Split Handshake Session Establishment. The TOE keeps state about connections or pseudo-connections and uses the information to permit or drop information flow. The TOE permits information flow between two subjects (i.e., from the physical interface on which network traffic entered to the physical interface determined by the destination address in the network packet) only where a security policy is defined between the source and destination zones that includes a rule that grants permission, based on the information security attributes listed above and the corresponding settings in the policy rule. A security policy rule includes the following attributes against which network packets can be compared: ● Source Zone, Destination Zone—zones must be of the same type (Layer 2, Layer 3, or Virtual Wire). Multiple zones can be specified in a single rule to simplify management ● Source Address, Destination Address—the IPv4 or IPv6 addresses for which the rule applies. Addresses must first be defined by the administrator, who specifies a name for the address and the actual IPv4 or IPv6 addresses to be associated with that name. Addresses can be specified as a single address, an address with a mask, or an address range. Addresses can also be combined into address groups to simplify management ● Service—specifies services to limit applications to specific protocols and port numbers. A security policy rule also includes the following attributes that determine what the TOE does with the network packet: ● Action—can be ‘allow’ or ‘drop’ ● Profiles—specifies any checking to be performed by the security profiles such as IPsec crypto Security and IKE Network Security. These profile allow/require the network traffic to be PROTECTed.) ● Options—specifies the following additional processing options for network packets matching the rule:

	○ Log Setting—generate log entries in the local traffic log ○ Schedule—limits the days and times when the rule is in effect (e.g., an 'allow' rule might be active only during normal business hours) ○ QoS Marking—change the Quality of Service (QoS) marking on packets matching the rule ○ Disable Server Response Inspection—disables packet inspection from the server to the client, which may be useful under heavy server load conditions. Prior to matching packets with the policy rules, fragmented packets are reassembled. Upon receiving a packet that is not associated with an established session (a packet with the SYN flag set without a corresponding ACK flag being set), the packet will be matched to the security rules to make a determination of whether to allow or drop the information flow. If the packet is associated with an established session (packet sequence number, acknowledgment number, and flags match an existing session record), the information flow is permitted. The Security Administrator may limit the number of half-open TCP connections and defines the thresholds that constitute flooding. The DoS Protection profile sets the thresholds at which the firewall generates a DoS alarm, takes action such as Random Early Drop, and drops additional incoming connections. The dropped connections are logged, if configured so, but are always counted. A DoS Protection profile policy rule that is set to protect (rather than to allow or drop packets) determines the criteria for packets to match (such as source address) in order to be counted toward the thresholds. The DoS Protection policy counts all connection attempts toward the thresholds. This flexibility permits the blacklisting certain traffic, or whitelist certain traffic and treat other traffic as DoS traffic. When the incoming rate exceeds the maximum threshold, the firewall blocks incoming traffic from the source address. The TOE uses a patented technology called App-ID to identify and control applications based on knowing exactly what the application is by evaluating the content of the traffic. This unique approach to traffic classification allows the TOE to provide visibility and control of the actual application, besides the ports or protocols that are allowed. App-ID is session "state" aware which allows the TOE to allow or block subsequent packets in a session. The TOE maintains a session "state" table for all sessions as part of the traffic processing layer of the device. If a packet doesn't match an existing session, then it is forced through the policy lookup process to determine if it should be allowed or not. If allowed, a session will be created. The logging can be enabled as well. The application decoder builds the state table based on the relevant RFCs. The TOE creates dynamic rules, maintaining the session states to support processing the FTP network protocol traffic for TCP data sessions in accordance with the FTP protocol as specified in RFC 959 using the FTP App-ID. The FTP App-ID identifies the application based on its unique properties and transaction characteristics using the App-ID technology to dynamically open pinholes to establish the connection, determine the parameters for the session and negotiate the ports that will be used for the transfer of data; these applications use the application-layer payload to communicate the dynamic TCP ports on which the application opens data connections. For such applications, the firewall serves as an Application Level Gateway (ALG), and it opens a pinhole for a limited time and for exclusively transferring data or control traffic. The dynamic sessions are removed when the FTP sessions are terminated (i.e., closed) by the FTP server or client, or when the TCP timeout expires (i.e., maximum time session can be open without any activities). Logging can be enabled in the security policy rule configured to monitor the

	FTP traffic, and can log FTP session starts, FTP session ends, or both. It does not log individual FTP command or data packet. The device provides a setting such that the Security Administrator can enable or disable ICMP and SNMP for all users. The TOE rejects requests for access or services when received on an interface that is not associated with the source address from which the information flow is sourced (by administrator configured "Strict IP address check" in the Zone Protection Profile"). Traffic is dropped if the source address of the incoming traffic correspond to the IP address of an external broadcast network or loopback network; if the incoming traffic is received from the external network but has a source address that correspond to the internal network; or if traffic is received from the internal network but has a source address that correspond to the external network. The TOE rejects packets where the source address is equal to the address of the network interface where the network packet was received. Access or service requests are also rejected when the presumed source identity specifies a broadcast identity or a loopback identifier. Security rules to block, permit or log are applied to multicast traffic. The TOE rejects and logs packets where the source address of the network packet is defined as being on a multicast network. The TOE discards and logs strict source routing, loose source routing, and record route packets. The TOE blocks RFC 6598 "Carrier Grade NAT" IP address block of 100.64.0.0/10. In addition, requests in which the information received contains the set of host network identifiers by which information is to travel from the source subject to the destination subject are rejected. The TOE has the capability to block the following IPv6 traffic: ● block both inbound and outbound IPv6 Site Local Unicast addresses (FEC0::/10) ● block IPv6 Jumbo Payload datagrams (Option Type 194). ● drop all inbound and outbound IPv6 packets containing a Hop-by-Hop header with option type values intended for Destination Options ● drop all inbound IPv6 packets for which the layer 4 protocol and ports (undetermined transport) cannot be located. ● drop all inbound IPv6 packets with a Type 0 Routing header. ● drop all inbound IPv6 packets with a Type 1 or Types 3 through 255 Routing Header. ● drop all inbound IPv6 packets containing undefined header extensions/protocol values. ● drop fragmented IPv6 packets when any fragment overlaps another. ● drop all inbound IPv6 packets containing more than one Fragmentation Header within an IP header chain. ● block IPv6 multicast addresses (FF00::/8) as a source address. Following is a more detailed description of the TOE's firewall capability. When the TOE receives a packet, it first determines if it represents a new connection or if it is part of an existing session. If it is part of an existing session, the traffic is processed based on the parameters of the existing session. If it is a new connection, the TOE retrieves the source and destination zones and performs an initial policy lookup. If a policy is defined for the zone pair (i.e., source and destination zones) a session is created and packet processing proceeds. By default, traffic between each pair of security zones is blocked until at least one rule is added to allow traffic between the two zones. Sessions are not created for a new connection if there is no policy defined for the zone pair; or if

there is an initial deny rule for the application service (i.e. service-HTTP, service-https) matching the traffic with no applications defined.

The TOE performs the following steps when processing traffic:

- The traffic is passed through the Application Identification and Application Decoders to determine what type of application is creating the session.
- Once the application is known, the TOE performs a policy lookup with the following information:
    - The source/destination IP address
    - The source/destination security zone
    - The application and service (port and protocol, Next Header)
    - The source user[13] (when available)
    - If a security policy is found, the policy rules are compared against the incoming traffic in sequence and the first rule that matches the traffic is applied. If a policy rule matching all of the traffic attributes listed above is not found, or if it is found and it specifies a drop action, then the packet is dropped (or DISCARDed) and the session is deleted.
    - If the application flow is allowed and no further security profiles are applied then it is forwarded (it is allowed to BYPASS the tunnel).
    - If the application is allowed and there are additional security profiles set, it will be sent to the stream signature processor. The traffic matching the IPsec crypto Security profile would then flow through the IPsec tunnel and be classified as "PROTECTED".
        - If there is no SA that the IPsec can use to protect this traffic to the peer, IPsec uses IKE to negotiate with the remote peer to set up the necessary IPsec SAs on behalf of the data flow. The negotiation uses information specified in the IKE Network Profiles.

Security policies can also specify security profiles that may be used to protect against viruses, spyware, and other threats after the connection is established.

Security Profiles

Each security policy can include specification of one or more security profiles, which provide additional protection and control. Security profiles are configured and applied to firewall policy. Each security policy can specify one or more of the following security profiles:

- IPsec Crypto Security profile
- IKE Crypto Security profile

The TOE can remove existing traffic flows from the set of established traffic flows based on the session inactivity timeout and completion of the expected information flow for each protocol (TCP/UDP/ICMPv4/ICMPv6). The timeout period due to inactivity is administrator configurable from 1 – 6044800 seconds per protocol. Session removal becomes effective before the next packet that might match the session is processed.

The TOE implements an implicit "deny-all" rule to interfaces where a traffic filtering rule has been applied. If a policy rule matching all of the traffic attributes described is not found, or if it is found and it specifies a deny action, then the packet is dropped, and the

[13] Source user in policies is not within the scope of the evaluation.

	session is deleted. Session removal becomes effective before the next packet that might match the session is processed. The PAN-OS performs Strict IP Address check, reject, and is capable of logging network packets where the source or destination address of the network packet is defined as being an address "reserved for future use" as specified in RFC 5735 for IPv4. The administrator may also configure the TOE to reject and log network packets where the source or destination address of the network packet is defined as a link-local address, an "unspecified address" or an address "reserved for future definition and use" as specified in RFC 3513 for IPv6. The TOE rejects and is capable of logging invalid and fragmented IP packets which cannot be re-assembled completely. The TOE detects all invalid fragmented packets, such as a fragmented packet that partially overlaps a previously received fragment, or a fragmented packet with invalid length, and drops and/or logs them as configured in the Zone Protection Profiles. Optionally, the TOE can be configured to consider any fragmented packet as invalid and to drop and log them. IP fragments will be parsed, be reassembled by defragmentation process and fed back to parser starting with IP header. A fragment may be discarded due to tear-drop attack (overlapping fragments). The network traffic can go through the TOE only if the Policy Enforcement Module is fully functional and it is enforcing all policies. During start-up and initialization, the TOE runs a series of system checks and the FIPS power up self- tests to ensure the system is functioning correctly. If these tests run successfully, the TOE will bring up the control plane and data-plane system modules. The Policy Enforcement Module (running on Data Plane) uses the policy configuration information created from the Management Server Module (running on the control plane). The configuration information includes all of the policies required by the Policy Enforcement Module. Policies are used to control information flow on the network. Only once the Policy Enforcement Module running on the data-plane is up and running and the TOE's system configuration is applied to enforce all security policies, can the TOE pass the traffic. The TOE implements the following safeguards that prevent packets from flowing through the TOE without applying the ruleset in the event of a component failure. The traffic can go through the TOE only if the Policy Enforcement Module is fully functional and enforcing all policies as described above. The Policy Enforcement Module can be configured to stop traffic when the traffic or system logs are full. Whenever a failure occurs within the TOE that results in the TOE ceasing operation, the TOE securely disables its interfaces to prevent the unintentional flow of any information to or from the TOE and reloads. The Policy Enforcement Module uses the policy configuration information created from the Management Server Module. The configuration information includes all of the policies required by the Policy Enforcement Module. Policies are used to control information flow on the network.

6.10 Packet Filtering

FPF_RUL_EXT.1	The Packet Filtering function is a subset of the Stateful Traffic Filtering function. This section provides a brief overview and summary of the packet filtering function. The Stateful Traffic Filtering function Section 6.9 contains additional detail relevant to this function. On the TOE, security policies determine whether to block or allow a session based on traffic attributes such as the source and destination security zone, the source and

	destination IP address, protocol (as specified in RFCs), Next Header, the application, user, and the service. All traffic passing through the firewall is matched against a session and each session is matched against a security policy. When a session match occurs, the security policy is applied to bi-directional traffic (client to server and server to client) in that session. For traffic that doesn't match any defined rules, a final configurable deny or allow rule is applied. The default rules allow all intrazone (within the zone) traffic and deny all interzone (between zones) traffic. Typically, intrazone traffic is considered to be trusted however both intrazone and interzone traffic can be configured to deny all traffic if there is no rule match by clicking on the security policy and clicking on the Override button on the bottom on the Policy ->Security screen. In the evaluated configuration, the default deny all rule for interzone traffic should not be modified. Each rule can be configured to generate a log record when the traffic matches the defined rule using the 'policy->Security->options' selection. The logging option can be configured to log at the start of a session, or at the end of a session or both. Security policies are evaluated left to right and from top to bottom in a packet filtering table format. A packet is matched against the first rule that meets the defined criteria; after a match is triggered the subsequent rules are not evaluated. Therefore, the more specific rules must precede more generic ones in order to enforce the best match criteria. Traffic that matches a rule generates a log entry of the session in the traffic log, if logging is enabled for that rule. The TOE will drop the packets if one of its interfaces is overwhelmed by network traffic. The 7000 series provides higher performance, in order to compensate the FPGA is designed to drop IPv6 with "zero" destination in the initial ingress packet processing. This event is logged in the FPGA counter log "flow_fpga_rcv_igr_IPV6DSTZERO". The security policy rules that determine whether a packet is transferred from one interface to another is based on: 1. IP address of source as defined as the original IP address in the packet. 2. IP address of destination as defined as the original IP address in the packet. 3. Service used allows Layer 4 selection (TCP or UDP) port for the application. 4. Source Zone from which the traffic originates. 5. Destination Zone at which the traffic terminates.

6.11 Intrusion Prevention

IPS_ABD_EXT.1	The TOE supports anomaly-based traffic detection via configurable, correlation baseline using attributes in packets such as IP addresses, ports, protocols, and data payload. For example, the Security Administrator can configure an FTP login brute-force attempts detection and if the attempt threshold exceeds 10 attempts in 60 seconds (configurable) from the same source and/or destination address pair, then it will trigger the configured action. This frequency detection settings can be applied to any unique event (e.g., brute-force, Denial of Service, flooding attack, etc.). The Security Administrator can manually configure frequency detection on security events that can trigger an action (Allow, Alert, Reset Client, Reset Server, Reset Both, or Drop) from the Vulnerability/Threat signature rule based on the defined attributes. This feature is part of the Vulnerability Protection profile security policy which must have a Threat Prevention license. The Vulnerability profile is tied to a security policy rule with logging options which can be associated with

	source and destination zones. The security zones are tied to any network data interface (not management interface).
IPS_IPB_EXT.1	The TOE supports security rules based known-good or known-bad list rules mechanism, or an External Dynamic List (EDL) mechanism. The EDL must be referenced in the security policy rule with logging options or profile. EDL supports predefined or custom IP address list which is an address list of IP addresses (single or range), URLs, or domains. For the scope of this evaluation, we will focus on IP addresses only as URLs and domains are just IP addresses translated via DNS. We also support multiple EDLs in a security policy rule. When multiple lists are referenced, the Security Administrator can prioritize the order of evaluation to make sure the most important EDLs are checked first. The EDL can be defined externally and downloaded into the TOE, or built-in EDLs with an active Threat Prevention license. The Security Administrator cannot modify the contents of the bult-in lists but they can create custom EDL of their own. As administrator view the entries of an external dynamic list, the Security Administrator can exclude up to 100 entries from the list (i.e., known-good IP addresses). The TOE does not support filtering based on MAC addresses. The CC AGD will provide more detailed instructions on configuring security policy rules and EDLs.
IPS_NTA_EXT.1	The TOE supports multiple network interface types from Tap, Virtual Wire, Layer 2, and Layer 3 interfaces. The TOE also supports a dedicated management (MGMT) port to be connected to an isolated management network as well. ● Tap[14] ("Promiscuous") - A network tap is a device that provides a way to access data flowing across a computer network. Tap mode deployment allows administrator to passively monitor traffic flows across a network by way of a switch SPAN or mirror port. No IP address is required to be configured. ● Virtual Wire ("Bump in the Wire") – The virtual wire logically connects the two interfaces; hence, the virtual wire is internal to the TOE. No IP address is required to be configured. ● Layer 2 - In a Layer 2 deployment, the TOE provides switching between two or more networks. No IP address is required to be configured. ● Layer 3 ("Inline") - In a Layer 3 deployment, the TOE routes traffic between multiple ports. The network traffic traverses the TOE and is routed to the appropriate port. IP addresses are required on both ingress and egress ports. The physical medium of the port can be Gigabit Ethernet or Small Form-Factor Pluggable (SFP) which enables Ethernet or fiber cables to be connected. The different physical medium only affects the speed, performance, and latency of the bits transmitted and received, and has no security relevancy. The Security Administrator can configure the IPS policies order, but the default packet processing flow is described below. A packet is received from the ingress network interface. The TOE parses the packet and determines whether the packet is subject to further inspection[15]. If so, the TOE will continue with a session lookup after the packet enters the security processing stage. During the processing stage, the TOE may discard the packet due to protocol or security violation. In certain cases, due to the TOE attack prevention features (part of the Zone Protection profile), it will discard packets with or without any configurable options. Next, in the session lookup stage, the TOE will determine if the packet is part of an existing or new session. If it is part of an existing session, the TOE will determine if content ('payload') inspection is required (per security policy rule). If it is not part of an existing session, then the firewall (i.e., security policy) rules will be applied (in order) to determine if the packet will be dropped or a new session will be set up. If the packet is

[14] In the evaluated configuration, the TOE should be deployed inline to meet both firewall and IPS requirements in at least one interface. It is possible to configure some interfaces in inline and other in non-inline.
[15] If the packet is part of a VPN tunnel, it must be decrypted first before packet processing can proceed.

	allowed by the security policy rule, the session will be added to the allowed session table and the TOE will determine if content inspection will be applied based on the IPS policy configured per security policy rule with logging options. If content inspection is configured (per security policy rule) and the TOE is in inline deployment, any match of the signature rules with action set to drop will cause the packet to be discarded. If content inspection is not configured or no signature rule matches, then the packet will be sent to the Forward/Route stage where it will be sent out the appropriate egress network interface. The TOE supports three (3) types of rules: security policy rule (i.e., firewall rule), L3 & L4 header rule, and Vulnerability/Threat signature rule. The default precedence order is L3 & L4 header rule (includes flooding protection), security policy rule (includes IP List), and Vulnerability/Threat signature rule (include signature-based and anomaly-based rules). Just like with any rule, the rules can be ordered from top to down and the first match will trigger the configured action. The TOE supports all protocols (IPv4, IPv6, ICMPv4, ICMPv6, TCP, and UDP) on all the network data interfaces. The vendor has taken the TOE through extensive third-party interoperability testing (e.g., DoDIN-APL, ICSA, NSS Labs) and protocol compliance testing (e.g., USGv6).
IPS_SBD_EXT.1	The TOE supports custom and predefined layer 3 and layer 4 threat signature rules with logging options based on header and payload fields. Please refer to the CC AGD [CCECG] for complete signature rule composition but some important elements are rule name, unique threat identifier, packet capture, exempt IP, log severity, log interval, action, signature, pattern match, etc. The signature element includes its name, an optional description, its scope (full session or single transaction), and signature matching conditions. If pattern matching is specified, it also includes a pattern matching context (FTP, HTTP, SMTP) and signature pattern (defined using a regular expression). The TOE performs stream reassembly to detect malicious payload split across multiple non-fragmented packets, and this capability does not require any configuration. The signature rules are assigned to a Zone Protection profile which is then associated with a Security Zone. Security Zone is then applied to a network data interface (e.g., ethernet1/1). The TOE supports string-based exact matching in header and payload via custom and predefined vulnerability signature rules. Every header field in the IP/ICMP/TCP/UDP packets can be inspected. The vulnerability signature rules are assigned to a Vulnerability Protection profile. The Vulnerability Protection profile is then associated with a security policy rule or rules. Each vulnerability signature rule must have an action (Allow, Alert, Reset Client, Reset Server, Reset Both, or Drop) defined and will trigger if the signature matches the security policy allowed traffic. IPS_SBD_EXT.1.3 and IPS_SBD_EXT.1.4 (flooding a host, flooding a network, protocol and port scanning) attacks are processed as part of the Zone Protection profile prior to the security policy rule. In the Zone Protection profile, you can configure the action the same way as the security policy rule. Some attacks are always dropped such as land, ping of death, teardrop, IP spoof, MAC spoof, ICMP fragment attacks, etc. These are tracked with counters per network interface.

7. Protection Profile Claims

This ST is conformant to the [CFG_NDcPP-FW-VPNGW_V2.0], [NDcPP], [FW-Module], [VPNGW-Module], [SSHPKG], and [IPS-Module].

8. Rationale

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