
Palo Alto Networks PA-200 Series, PA-500, PA-800 Series, PA-3000 Series, PA-5000 Series, PA-5200 Series, PA-7000 Series, and VM Series Next-Generation Firewall with PAN-OS v8.0.6

Security Target

Version 1.0
16 March 2018

Prepared for:



Palo Alto Networks, Inc.
3000 Tannery Way
Santa Clara, CA 95054

Prepared by:



Accredited Testing and Evaluation Labs
6841 Benjamin Franklin Drive
Columbia, MD 21046

Table of Contents

1. SECURITY TARGET INTRODUCTION	1
1.1 SECURITY TARGET, TOE AND CC IDENTIFICATION	1
1.2 CONFORMANCE CLAIMS	3
1.3 CONVENTIONS	5
1.3.1 Terminology	5
1.3.2 Acronyms	5
2. PRODUCT AND TOE DESCRIPTION	7
2.1 INTRODUCTION	7
2.2 PRODUCT OVERVIEW	7
2.3 TOE OVERVIEW	10
2.4 TOE ARCHITECTURE	11
2.4.1 Physical Boundaries	13
2.4.2 Logical Boundaries	19
2.5 TOE DOCUMENTATION	21
3. SECURITY PROBLEM DEFINITION	22
4. SECURITY OBJECTIVES	23
4.1 SECURITY OBJECTIVES FOR THE OPERATIONAL ENVIRONMENT	23
5. IT SECURITY REQUIREMENTS	24
5.1 EXTENDED REQUIREMENTS	24
5.2 TOE SECURITY FUNCTIONAL REQUIREMENTS	26
5.2.1 Security Audit (FAU)	27
5.2.2 Cryptographic Support (FCS)	30
5.2.3 User Data Protection (FDP)	37
5.2.4 Identification and Authentication (FIA)	37
5.2.5 Stateful Traffic Filtering (FFW)	38
5.2.6 Security Management (FMT)	40
5.2.7 Protection of the TSF (FPT)	42
5.2.8 TOE Access (FTA)	43
5.2.9 Trusted Path/Channels (FTP)	43
5.3 TOE SECURITY ASSURANCE REQUIREMENTS	45
6. TOE SUMMARY SPECIFICATION	46
6.1 SECURITY AUDIT	46
6.2 CRYPTOGRAPHIC SUPPORT	47
6.3 USER DATA PROTECTION	54
6.4 IDENTIFICATION AND AUTHENTICATION	55
6.5 SECURITY MANAGEMENT	57
6.6 PROTECTION OF THE TSF	58
6.7 TOE ACCESS	60
6.8 TRUSTED PATH/CHANNELS	61
6.9 STATEFUL TRAFFIC FILTERING	61
7. PROTECTION PROFILE CLAIMS	67
8. RATIONALE	70

8.1 TOE SUMMARY SPECIFICATION RATIONALE.....70

LIST OF TABLES

Table 1 TOE Platforms15
Table 2 TOE Security Functional Components26
Table 3 Auditable Events27
Table 4 Assurance Components45
Table 5 Cryptographic Functions47
Table 6 FIPS 186-4 Conformance49
Table 7 Private Keys and CSPs49
Table 8 HMAC Key Length, Block Size, and Output Length51
Table 9 SFR Protection Profile Sources67
Table 10 Security Functions vs. Requirements Mapping.....71

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 v8.0.6.

The next-generation firewall includes the PA-200, PA-220, PA-500, PA-820, PA_850, PA-3020, PA-3050, PA-3060, PA-5020, PA-5050, PA-5060, PA-5220, PA-5250, PA-5260, PA-7050, and PA-7080 appliances and the virtual appliances in the VM-Series VM-50, VM-100, VM-200, VM-300, VM-500, VM-700, VM-1000-HV 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 SSL encryption.

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

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] and [FWcPP] 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, and specifies CAVP-validated cryptographic mechanisms.

The Security Target contains the following additional sections:

- Product and TOE 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-200 Series, PA-500, PA-800 Series, PA-3000 Series, PA-5000 Series, PA-5200 Series, PA-7000 Series, and VM Series Next-Generation Firewall with PAN-OS v8.0.6 Security Target

ST Version – Version 1.0

ST Date – 16 March 2018

TOE Identification – Palo Alto Networks PA-200 Series, PA-500, PA-800 Series, PA-3000 Series, PA-5000 Series, PA-5200 Series, PA-7000 Series, and VM Series, Next-Generation Firewall with PAN-OS v8.0.6. The specific Firewall appliance models include:

1. PA-200 Series
 - a. PA-200
 - b. PA-220
2. PA-500
3. PA-800 Series
 - a. PA-820
 - b. PA-850

4. PA-3000 Series
 - a. PA-3020
 - b. PA-3050
 - c. PA-3060
5. PA-5000 Series
 - a. PA-5020
 - b. PA-5050
 - c. PA-5060
6. PA-5200 Series
 - a. PA-5220
 - b. PA-5250
 - c. PA-5260
7. PA-7000 Series
 - a. PA-7050
 - b. PA-7080
8. VM-Series - VM-Series
 - a. VM-1000-HV
 - b. VM-300
 - c. VM-200
 - d. VM-100
 - e. VM-50
 - f. VM-500
 - g. VM-700

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

- VMware
 - VMware ESXi with vSphere 5.1, 5.5, 6.0, or 6.5
- Linux KVM
 - CentOS/RedHat Enterprise Linux: 7.2.1511 (QEMU-KVM 1.5.3 and libvirt 2.0.0; Open vSwitch: 2.3.1 and later)
- Microsoft Hyper-V Server 2012 R2 ---- 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 2012 R2, or as an add-on/role for Windows Server 2012 R2.

Each VM-Series virtual appliance in its evaluated configuration is installed on a hardware platform that includes a VMware, Linux KVM, or Microsoft Hyper-V hypervisor and an Intel Core or Xeon processor based on the Ivy Bridge, Haswell, or Broadwell microarchitectures that implement Intel Secure Key.

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

VMware ESXi 5.5:

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

And

- PacStar PS451 Processor: Intel Xeon CPU E3-1258L v4

- Network Interfaces: Intel I218-LM:MGMT port-vmnic0-, Intel I210: vmnic 1-4 vvv

KVM:

- Dell PowerEdge R730 Server running on an Intel Xeon E5-2630 v3 (Haswell microarchitecture) with Broadcom 5720 NIC
- Memory: 64 GB ECC DDR4 2133

Microsoft Hyper-V:

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

TOE Developer – Palo Alto Networks, Inc.

Evaluation Sponsor – Palo Alto Networks, Inc.

CC Identification – *Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 4, September 2012*

1.2 Conformance Claims

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

- *collaborative Protection Profile for Stateful Traffic Filter Firewalls, Version 1.0, 27 February 2015* [FWcPP] with the optional SFR: FFW_RUL_EXT.2.1
- *collaborative Protection Profile for Network Devices, Version 1.0, 27 February 2015* [NDcPP] and including the following optional SFRs: FCS_HTTPS_EXT.1, FCS_TLSC_EXT.1, FCS_TLSC_EXT.2, FCS_TLSS_EXT.1 and FCS_TLSS_EXT.2. The following NIAP Technical Decisions apply to this PP and have been accounted for in the ST development and the conduct of the evaluation:
 - TD0090: NIT Technical Decision for FMT_SMF.1.1 Requirement in NDcPP
 - TD0094: NIT Technical Decision for validating a published hash in NDcPP
 - TD0095: NIT Technical Interpretations regarding audit, random bit generation, and entropy in NDcPP
 - TD0096: NIT Technical Interpretation regarding Virtualization
 - TD0111: NIT Technical Decision for third party libraries and FCS_CKM.1 in NDcPP and FWcPP
 - TD0112: NIT Technical Decision for TLS testing in the NDcPP v1.0 and FW cPP v1.0.
 - TD0113: NIT Technical Decision for testing and trusted updates in the NDcPP v1.0 and FW cPP v1.0
 - TD0114: NIT Technical Decision for Re-Use of FIPS test results in NDcPP and FWcPP
 - TD0115: NIT Technical Decision for Transport mode and tunnel mode in IPsec communication in NDcPP and FWcPP
 - TD0116: NIT Technical Decision for a Typo in reference to RSASSA-PKCS1v1_5 in NDcPP and FWcPP
 - TD0117 (supercedes TD0093): NIT Technical Decision for FIA_X509_EXT.1.1 Requirement in NDcPP
 - TD0125: NIT Technical Decision for Checking validity of peer certificates for HTTPS servers

- TD0126: NIT Technical Decision for TLS Mutual Authentication
- TD0130: NIT Technical Decision for Requirements for Destruction of Cryptographic Keys
- TD0143: NIT Technical Decision for Failure testing for TLS session establishment in NDcPP and FWcPP
- TD0151: NIT Technical Decision for FCS_TLSS_EXT Testing - Issue 1 in NDcPP v1.0.
- TD0153: NIT Technical Decision for Auditing of NTP Time Changes in the NDcPP v1.0 and FW cPP v1.0
- TD0154: NIT Technical Decision for Versions of TOE Software in the NDcPP v1.0 and FW cPP v1.0
- TD0155: NIT Technical Decision for TLSS tests using ECDHE in the NDcPP v1.0.
- TD0156: NIT Technical Decision for SSL/TLS Version Testing in the NDcPP v1.0 and FW cPP v1.0
- TD0160: NIT Technical Decision for Transport mode and tunnel mode in IPsec communications
- TD0168: NIT Technical Decision for Mandatory requirement for CSR generation
- TD0169: NIT Technical Decision for Compliance to RFC5759 and RFC5280 for using CRLs
- TD0170: NIT Technical Decision for SNMPv3 Support TD0181: NIT Technical Decision for Self-testing of integrity of firmware and software
- TD0181: NIT Technical Decision for Self-testing of integrity of firmware and software.
- TD0182: NIT Technical Decision for Handling of X.509 certificates related to ssh-rsa and remote comms
- TD0184: NIT Technical Decision for Mandatory use of X.509 certificates
- TD0185: NIT Technical Decision for Channel for Secure Update
- TD0186: NIT Technical Decision for Applicability of X.509 certificate testing to IPsec
- TD0187: NIT Technical Decision for Clarifying FIA_X509_EXT.1 test 1
- TD0188: NIT Technical Decision for Optional use of X.509 certificates for digital signatures
- TD0199: NIT Technical Decision for Elliptic Curves for Signatures
- TD0223: NIT Technical Decision for "Expected" vs "unexpected" DNs for IPsec Communications
- TD0224: NIT Technical Decision Making DH Group 14 optional in FCS_IPSEC_EXT.1.11
- TD0225: NIT Technical Decision for Make CBC cipher suites optional in IPsec
- TD0226: NIT Technical Decision for TLS Encryption Algorithms
- TD0227: NIT Technical Decision for TOE acting as a TLS Client and RSA key generation
- TD0228: NIT Technical Decision for CA certificates - basicConstraints validation
- TD0235: NIT Technical Decision adding DH group 14 to the selection in FCS_CKM.2
- TD0255: NIT Technical Decision for TLS Server Tests - Issue 3: Verification of application of encryption
- TD0256: NIT Technical Decision for Handling of TLS connections with and without mutual authentication
- TD0257: NIT Technical Decision for Updating FCS_DTLSC_EXT.x.2/FCS_TLSC_EXT.x.2 Tests 1-4

- TD0262: NIT Technical Decision for TLS server testing - Empty Certificate Authorities list.
- Common Criteria for Information Technology Security Evaluation Part 2: Security functional components, Version 3.1, Revision 4, September 2012.
 - Part 2 Extended
- Common Criteria for Information Technology Security Evaluation Part 3: Security assurance components, Version 3.1 Revision 4, September 2012.
 - Part 3 Conformant.

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.
 - Iteration: allows a component to be used more than once with varying operations. In the ST, iteration is indicated by a number in parentheses placed at the end of the component. For example FDP_ACC.1 (1) and FDP_ACC.1 (2) indicate that the ST includes two iterations of the FDP_ACC.1 requirement, (1) and (2).
 - Assignment: allows the specification of an identified parameter. Assignments are indicated using bold and are surrounded by brackets (e.g., **[assignment]**). Note that an assignment within a selection would be identified in italics and with embedded bold brackets (e.g., **[*selected-assignment*]**).
 - Selection: allows the specification of one or more elements from a list. Selections are indicated using bold italics and are surrounded by brackets (e.g., **[*selection*]**).
 - Refinement: allows the addition of details. Refinements are indicated using bold, for additions, and strike-through, for deletions (e.g., "... **all** objects ..." or "... ~~some~~ **big** things ..."). Note that 'cases' that are not applicable in a given SFR have simply been removed without any explicit identification.
- Other sections of the ST – Other sections of the ST use bolding to highlight text of special interest, such as captions.
- The ST does not highlight operations that have been completed by the PP and EP authors.

1.3.1 Terminology

The following terms and abbreviations are used in this ST:

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, and the IKE Network 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

AES	Advanced Encryption Standard
CBC	Cipher-Block Chaining
CC	Common Criteria for Information Technology Security Evaluation

CEM	Common Evaluation Methodology for Information Technology Security
CM	Configuration Management
CLI	Command Line Interface
CPU	Central Processing Unit
DH	Diffie-Hellman
EEPROM	Electrically Erasable Programmable Read-Only Memory
EP	Extended Package
FIA	Identification and Authentication CC Class
FIPS	Federal Information Processing Standard
FMT	Security Management CC Class
FSP	Functional Specification
FTP	File Transfer Protocol
GUI	Graphical User Interface
GB	Gigabyte
HMAC	Hashed Message Authentication Code
HTTP(S)	Hypertext Transfer Protocol (Secure)
IKE	Internet Key Exchange
IP	Internet Protocol
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
IPsec	Internet Protocol Security
NDPP	Protection Profile for Network Devices
NAT	Network Address Translation
NIST	National Institute of Standards and Technology
PP	Protection Profile
QoS	Quality of Service
REST	Representational State Transfer
RSA	Rivest, Shamir and Adleman (algorithm for public-key cryptography)
SA	Security Association
SAR	Security Assurance Requirement
SFP	Security Function Policy
SFR	Security Functional Requirement
SHA	Secure Hash Algorithm
SM	Security Management
SMR	Security Management Roles
SMTP	Simple Mail Transfer Protocol
SNMP	Simple Network Management Protocol
SSH	Secure Shell
SSL	Secure Socket Layer Protocol
ST	Security Target
TB	Terabyte
STFF	Stateful Traffic Filter Firewall (EP)
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Functions
TSP	TOE Security Policy
UDP	User Data Protection
URL	Uniform Resource Locator
VLAN	Virtual Local Area Network
VM	Virtual Machine

2. Product and TOE Description

The TOE comprises the range of Palo Alto Networks hardware and virtual next-generation firewall devices running PAN-OS v8.06. This section first provides an overview of the capabilities of the next-generation firewall product, and then proceeds to describe the TOE itself, providing an overview of the evaluated capabilities and descriptions of the TOE architecture and physical and logical boundaries.

2.1 Introduction

Palo Alto Networks provides a wide suite of enterprise-level next-generation firewalls, with a diverse range of security features for the enterprise network.

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 following three paragraphs describe components (Panorama, Wildfire, GlobalProtect) that can be deployed in the operational environment of the TOE, but are outside the TOE boundary.

Panorama network security management enables control of a distributed network of Palo Alto firewalls from one central location. An administrator may view all of the firewall traffic, manage all aspects of device configuration, push global policies, and generate reports on traffic patterns or security incidents — all from a single console. The use of Panorama to manage Palo Alto firewalls has not been evaluated—the evaluated configuration comprises individual firewalls managed in isolation, not a distributed solution.

The WildFire appliance provides an on-premises WildFire private cloud, enabling the analysis of suspicious files in a sandbox environment without requiring the firewall to send files out of network. The WildFire appliance can be configured to host a WildFire private cloud where the firewall is configured to submit samples to the local WildFire appliance for analysis. The WildFire appliance can be configured to locally generate antivirus and DNS signatures for discovered malware, and to assign a URL category to malicious links. The use of Wildfire to operate with Palo Alto firewalls has not been evaluated.

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 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. By eliminating the blind spots in mobile workforce traffic, the organization maintains a consistent view into applications. The use of Global Protect to operate with Palo Alto firewalls has not been evaluated.

2.2 Product Overview

This sub-section describes capabilities of the Palo Alto Networks next-generation firewall products. It should be noted that many of these capabilities are not covered within the scope of the evaluation. The scope of the evaluation is covered in the subsequent sub-sections that provide the TOE overview and describe the TOE architecture and physical and logical boundaries.

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 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.

- Secure Socket Layer (SSL) decryption policies, the SSL decryption feature uses an SSL proxy to establish itself as a man-in-the-middle proxy, which decrypts and controls the traffic within the SSL tunnel that traverses the TOE. SSL decryption is configured as a rulebase in which match criteria include zone, IP address, and User-ID.
- SSH Decryption is checked using the SSH application signature, a policy lookup will occur on the decrypt rule to see if this session should be decrypted. If yes, the TOE will set up a man-in-the middle to decrypt the session and decide if any port-forwarding request is sent in that session. As soon as any port forwarding is detected, the application becomes an SSH-tunnel, and based on the policy, the session might get denied.
- Application Override policies
- User Identification Agent (UIA) policy enforcement - provides the firewall with the capability to automatically collect user-specific information, and provides mapping information between IP addresses and network users, that is used in security policy enforcement and reporting. 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:

- Antivirus profiles
- Antispyware profiles
- Vulnerability Protection profiles
- File blocking profiles
- URL filtering profiles
- Data Filtering profiles
- DoS Protection profiles
- IPsec crypto Security profiles
- IKE Network 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.
- Threat prevention — the firewall includes threat prevention capabilities that can protect the network from viruses, worms, spyware, and other malicious traffic.
- Traffic visibility — the firewall includes the capability to generate extensive reports, logs, and notification mechanisms that provide detailed visibility into network application traffic and security events.

- Fail-safe operation — the firewall can be configured for fault-tolerant operations, where the firewall can be deployed in active/passive pairs so that if the active firewall fails for any reason, the passive firewall becomes active automatically with no loss of service.
- Management — each firewall can be managed through a Graphical User Interface (GUI). 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 and reporting. The interface also provides dynamic visibility tools that enable views into the actual applications running on the network. The GUI can identify the applications with the most traffic and the highest security risks. When configured in a Common Criteria mode of operation, the GUI is secured using HTTP over TLS.

Firewall Policy Enforcement

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

- Application Protocol/Port: App-ID identifies the protocol (such as TCP or UDP) and the port number of the traffic.
- Application Protocol Decoding: App-ID's protocol decoders determine if the application is using a protocol as a normal application transport or if it is only using the apparent protocol to hide the real application protocol.
- Application Signatures: App-ID uses context-based signatures, which look for unique application properties and related transaction characteristics to correctly identify the application regardless of the protocol and port being used.
- Heuristics: App-ID requires multi-packet heuristics for identifying some encrypted applications like Skype and encrypted Bittorrent.

The application-centric nature of App-ID means that it cannot only identify and control traditional applications such as SMTP, FTP, and SNMP, but it can also accurately identify many more applications through the use of protocol decoders and application signatures.

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. The threat prevention capabilities have not been evaluated.

App-ID and Threat Prevention Signature Updates

App-ID signatures 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 instructed to contact Palo Alto Networks' update server to download new content updates as they are made available. The connection to the update server is secured with TLS using FIPS-approved algorithms. For an additional layer of protection, Palo Alto Networks has chosen to sign (using RSA-2048) and encrypt (using AES-256). Although secure communication between the TOE and the update server has been tested, the App-ID and Threat Prevention capabilities and their use of signatures is not in the scope of evaluation.

Management

The next-generation firewall provides both direct and remote connections for the Web Management interface. The Web interface provides administrators with the ability to manage, configure and monitor the TOE either through a direct connection or via HTTPS from a web browser.

User Identification Agent (UIA)

The UIA is software installed on one or more PCs in the operational environment on the protected network. The UIA provides the firewall with the capability to automatically collect user-specific information that is used in security policy enforcement and reporting. The UIA is not related to Identification and Authentication. Use of the UIA with the TOE was not covered in the scope of the evaluation.

Common Criteria Compliant Mode of Operation

The TOE is compliant with the capabilities outlined in this Security Target only when operated in Common Criteria mode. Common Criteria mode is a special operational mode in which the FIPS 140-2 requirements for startup and conditional self-tests as well as algorithm selection are enforced. In this mode, only FIPS-approved and FIPS-allowed cryptographic algorithms are available.

2.3 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-200, PA-220, PA-500, PA-820, PA-850, PA-3020, PA-3050, PA-3060, PA-5020, PA-5050, PA-5060, PA-5220, PA-5250, PA-5260, PA-7050, and PA-7080 appliances and the virtual appliances in the VM-Series VM-50, VM-100, VM-200, VM-300, VM-500, VM-700, VM-1000-HV with PAN-OS v8.0.6. The next-generation firewall provides policy-based application visibility and control to protect traffic flowing through the enterprise network.

The focus of this evaluation is on the TOE functionality supporting the claims in the *collaborative Protection Profile for Stateful Traffic Filter Firewalls* and the *collaborative Protection Profile for Network Devices*. have been applied to the Security Target to specify the minimum required capabilities. (See section 1.2 for specific version information).

The TOE is a stateful traffic filter firewall appliance. The scope of the evaluation does not cover Layer 2 switching, VLAN, or transparent in-line deployments.

The TOE provides control over the traffic allowed to access a protected network. The administrator defines security policies for specific applications. The security policy rules that determine whether a packet is transferred from one interface to another are 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. Application service (such as HTTP) limited to specific TCP and/or UDP port numbers.
4. Source Zone from which the traffic originates.
5. Destination Zone at which the traffic terminates.

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 TOE is able to generate logs of security relevant events. The logged audit records identify the date and time, the nature or type of the triggering event, an indication of whether the event succeeded or failed, and the identity of the user responsible for the event. The TOE stores the audit records locally and protects them from unauthorized deletion by allowing only users in the pre-defined Audit Administrator role to access the audit trail with delete privileges. The TOE can be configured to send generated audit records to an external Syslog server 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.

The TOE provides a GUI management interface to support security management of the TOE. The GUI is accessible via direct connection to the management port on the device, or remotely over HTTPS or IPsec. The management interfaces enable the authorized administrators to configure the TOE functions and to manipulate TOE data.

The TOE requires all administrators to be identified and authenticated before they can access any of the TOE functions. The administrator can logon to the GUI by using a secure connection (HTTPS) from a web browser. The administrator enters the IP address of the TOE and their username and password. Passwords can be composed of upper and lower

case letters, numbers and special characters. There are no restrictions on any password field character sets. The minimum password length is configurable by the administrator up to a maximum length of 31 characters.

The TOE also can be configured to require a client certificate (mutual authentication) and additionally require the username and password or not. In order for an administrator to login to the GUI using IPsec, an IPsec tunnel has to be established between the client laptop/management station and the TOE. The administrator uses a third party IPsec client for setting up an IPsec tunnel to the TOE. Regardless of whether a user logs in using an HTTPS or IPsec 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 on the certificate is validated by the TOE.

The TOE controls user access to commands and resources based on user role. Users are given permission to access a set of commands and resources based on their user role. The TOE has the following pre-defined custom administrator roles: auditadmin, cryptoadmin, and securityadmin. These administrator roles are all considered Security Administrator as outlined in the Protection Profiles.

The only capabilities allowed prior to users authenticating are the display of the warning banner before authentication. The TOE provides both local and remote users the ability to logout (or terminate) their sessions as directed by the user. The TOE can also be configured by an administrator to set an interactive session timeout value.

The TOE provides self-tests at start-up (which are also on-demand tests available to administrators) to demonstrate the correct operation of: key error detection, cryptographic algorithms, and RNG. Conditional self-tests are also run during the course of normal operation. The self-tests verify the integrity of stored TSF executable code and TSF data.

2.4 TOE Architecture

The TOE comprises two subsystems: the control plane and the 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.

PAN-OS implements two kernels, one for the data plane and one for each instance of the data plane. The data plane software always runs on a Cavium OCTEON CPU. The management plane software runs on an Intel CPU on all platforms except the PA-200, PA-220, PA-500, PA-820, and PA-850. The PA-200, PA-220, PA-500, PA-820, and PA-850 devices are slightly different, since they have only one, multicore Cavium CPU. On these devices, the management plane software runs on the Cavium along with the data plane software, but on different cores. VM devices are similar to the PA-200/500/800 devices. Depending on the number of cores allocated for PAN-OS, one core will be allocated for the management plane and one or more cores for data plane software.

The Cavium OCTEON CPUs are all based on the MIPS64 architecture and include a hardware random number generator (RNG), which provides the entropy source to the DRBG implementations on appliances. On platforms where the management plane and data plane run on separate CPUs, random data is fed back from the data plane into the management plane.

The following diagram depicts both the hardware and software architecture of the TOE. Note, the User Identification Agent is in the operational environment.

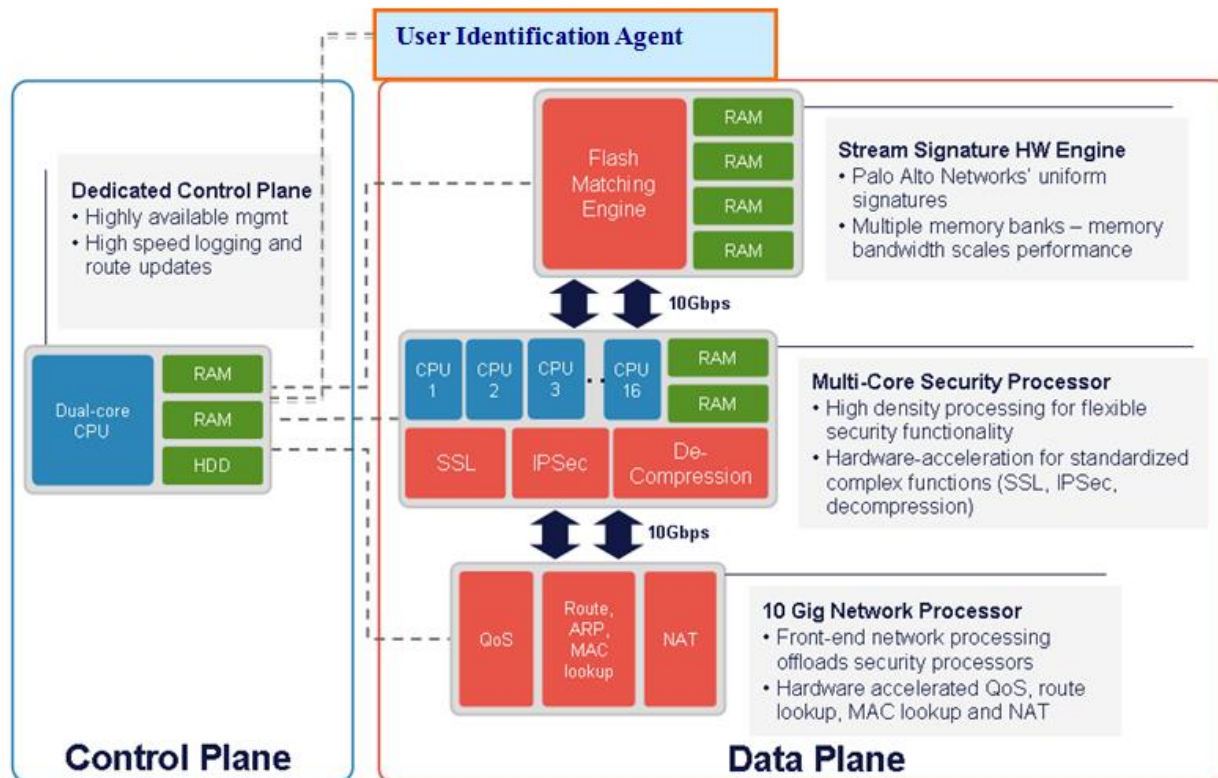


Figure 1: TOE Architecture

The control plane includes a dual 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

The control plane provides all device management functionality, including:

- All management interfaces – provide a both direct and remote connection for the Web Interface GUI.
- Configuration management of the device, such as controlling the changes made to the device configuration, as well as the compilation and pushing to the dataplane of a configuration change.
- Logging infrastructure for traffic, threat, alarm, configuration, and system logs.
- Reporting infrastructure for reports, monitoring tools, and graphical visibility tools
- 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.

Data Plane

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. This capability is used in the evaluated configuration to support network flows using FTP.

- Policy lookups to determine what security policy to enforce and what actions to take, such as packet logging.
- Logging, with all logs sent to the control plane for processing and storage.

VM-Series

The VM-Series on specified hardware supports the exact same firewall 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 Ivy Bridge, Haswell, or Broadwell microarchitectures that implement Intel Secure Key, and Network Interface Controllers supported by the Server.

2.4.1 Physical Boundaries

The TOE consists of the following components:

- Hardware appliance-includes the physical port connections on the outside of the appliance cabinet and its own internal clock which it uses to provide a reliable time source for audit records.
- Virtualized Firewalls installed on specified hardware - the VM-Series supports the exact same next-generation firewall features available in the physical form factor appliances, allowing an administrator to safely enable applications flowing into, and across your 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 v8.0.6 – 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 Appweb (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 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-200, PA-220, PA-500, PA-820, PA-850, PA-3020, PA-3050, PA-3060, PA-5020, PA-5050, PA-5060, PA-5220, PA-5250, PA-5260, PA-7050, and PA-7080); and the virtual appliances on specified hardware in the VM-Series VM-50, VM-100, VM-200, VM-300, VM-500, VM-700, VM-1000-HV. The next-generation firewall models differ in their performance capability, but they provide the same security functionality.

Virtual systems are supported by default (without an additional license) on the PA-220, PA-820, PA-850, PA-3020, PA-3050, PA-3060, PA-5020, PA-5050, PA-5220, PA-5250, PA-5060, PA-7050, and PA-7080. The PA-200 and the PA-500 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:

- PA-200: 8 RJ-45 10/100/1000 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).
- PA-220: 8 RJ-45 10/100/1000 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); 1 USB, and 1 Micro USB Console.

- PA-500: 8 RJ-45 10/100/1000 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).
- 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, and 1 Micro USB Console.
- 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, and 1 Micro USB Console.
- PA-3020/PA-3050: 12 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); 1 RJ-45 port for connecting a serial console (management console port); and 2 RJ-45 ports for high-availability (HA) control and synchronization
- PA-3060: 8 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); 1 RJ-45 port for connecting a serial console (management console port); and 2 RJ-45 ports for high-availability (HA) control and synchronization.
- PA-5020: 12 RJ-45 10/100/1000 ports for network traffic. 8 Small Form-Factor Pluggable (SFP) ports for network traffic. One RJ-45 port to access the device management interfaces through an Ethernet interface. One RJ-45 port for connecting a serial console. Two RJ-45 ports for high-availability (HA) control and synchronization.
- PA-5050: 12 RJ-45 10/100/1000 ports for network traffic. Eight Small Form-Factor Pluggable (SFP) ports for network traffic. Four SFP+ ports for network traffic. One RJ-45 port to access the device management interfaces through an Ethernet interface. One RJ-45 port for connecting a serial console. Two RJ-45 ports for high-availability (HA) control and synchronization.
- PA-5060: 12 RJ-45 10/100/1000 ports for network traffic. Eight Small Form-Factor Pluggable (SFP) ports for network traffic. Four SFP+ ports for network traffic. One RJ-45 port to access the device management interfaces through an Ethernet interface. One RJ-45 port for connecting a serial console. Two RJ-45 ports for high-availability (HA) control and synchronization.
- PA-5220: Four 100/1000/10G Cu, sixteen 1G/10G SFP/SFP+, four 40G QSFP+ for network traffic; Two RJ-45 port to access the device management interfaces through an Ethernet interface; One RJ-45 port for connecting a serial console, one 40G QSFP+ HA for high-availability (HA) control and synchronization.
- PA-5250: Four 100/1000/10G Cu, sixteen 1G/10G SFP/SFP+, four 40G/100G QSFP28 for network traffic; Two RJ-45 port to access the device management interfaces through an Ethernet interface; One RJ-45 port for connecting a serial console, one 40G/100G QSFP28 for high-availability (HA) control and synchronization.
- PA-5260: Four 100/1000/10G Cu, sixteen 1G/10G SFP/SFP+, four 40G/100G QSFP28 for network traffic; Two RJ-45 port to access the device management interfaces through an Ethernet interface; One RJ-45 port for connecting a serial console, one 40G/100G QSFP28 for high-availability (HA) control and synchronization.
- PA-7050: 12 gig copper ports for network traffic, eight Small Form-Factor Pluggable (SFP) ports for network traffic and four SFP+ ports for network traffic per blade OR two Quad Small Form-Factor Pluggable (QSFP) for network traffic per blade and twelve SFP+ ports for network traffic per blade (6 blades max). One RJ-45 port to access the device management interfaces through an Ethernet interface. One RJ-45 port for connecting a serial console. Two QSFP ports for high-availability (HA) control and synchronization.
- PA-7080: 12 gig copper ports for network traffic, eight Small Form-Factor Pluggable (SFP) ports for network traffic and four SFP+ ports for network traffic per blade OR two Quad Small Form-Factor

Pluggable (QSFP) for network traffic per blade and twelve SFP+ ports for network traffic per blade (10 blades max). One RJ-45 port to access the device management interfaces through an Ethernet interface. One RJ-45 port for connecting a serial console. Two QSFP ports for high-availability (HA) control and synchronization.

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

- A computer either directly connected or remotely connected to the appliance Management port via an RJ-45 Ethernet cable. The appliance Management port is an out-of-band management port that provides access to the GUI via HTTPS. The appliance Management port separates the management functions of the firewall from the data processing functions, safeguarding access to the firewall and enhancing performance. The computer is part of the operational environment and required to have a web browser (for accessing the GUI).



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 can include the following:






- syslog server,
- update server,
- Panorama appliance
- WildFire appliance
- Global Protect application
- Web browsers - Internet Explorer (IE, Release 7 and later, recommended IE Release 10 and later), Firefox (version 3.6 or later), Safari (version 5 or later), and Chrome (version 11 or later) browser.





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 2008 32-bit and 64-bit, Windows Server 2012, and Windows Server 2012 R2.

Table 1 TOE Platforms

Product Identification	Illustration	Description
PA-200		<ul style="list-style-type: none"> • 100 Mbps firewall throughput(App-ID enabled) • 50 Mbps threat prevention throughput • 64,000 max sessions • 1,000 new sessions per second • 10 security zones • 250 max number of policies
PA-220		<ul style="list-style-type: none"> • 500Mbps firewall throughput(App-ID enabled) • 150 Mbps threat prevention throughput • 64,000 max sessions • 4,200 new sessions per second • 15 security zones • 250 max number of policies

Product Identification	Illustration	Description
PA-500		<ul style="list-style-type: none"> • 250 Mbps firewall throughput (App-ID enabled) • 100 Mbps threat prevention throughput • 64,000 max sessions • 7,500 new sessions per second • 3 virtual routers • N/A virtual systems (base/max) • 20 security zones • 1,000 max number of policies
PA-820		<ul style="list-style-type: none"> • 1.9 Gbps firewall throughput (App-ID enabled) • 780 Mbps threat prevention throughput • 192,000 max sessions • 9,500 new sessions per second • 5 virtual routers • 40 security zones • 1,500 max number of policies
PA-850		<ul style="list-style-type: none"> • 1.9 Gbps firewall throughput (App-ID enabled) • 780 Mbps threat prevention throughput • 192,000 max sessions • 9,500 new sessions per second • 5 virtual routers • 40 security zones • 1,500 max number of policies
PA-3020		<ul style="list-style-type: none"> • 2 Gbps firewall throughput (App-ID enabled) • 1 Gbps threat prevention throughput • 250,000 max sessions • 50,000 new sessions per second • 10 virtual routers • 1/6 virtual systems (base/max) • 40 security zones • 2,500 max number of policies
PA-3050		<ul style="list-style-type: none"> • 4 Gbps firewall throughput (App-ID enabled) • 2 Gbps threat prevention throughput • 500,000 max sessions • 50,000 new sessions per second • 10 virtual routers • 1/6 virtual systems (base/max) • 40 security zones • 5,000 max number of policies

Product Identification	Illustration	Description
PA-3060		<ul style="list-style-type: none"> • 4 Gbps firewall throughput (App-ID enabled1) • 2 Gbps threat prevention throughput • 500,000 max sessions • 50,000 new sessions per second • 10 virtual routers • 1/6 virtual systems (base/max) • 40 security zones • 5,000 max number of policies
PA-5020		<ul style="list-style-type: none"> • 5 Gbps firewall throughput (App-ID enabled1) • 2 Gbps threat prevention throughput • 1,000,000 max sessions • 120,000 new sessions per second • 20 virtual routers • 10/20 virtual systems (base/max) • 80 security zones • 10,000 max number of policies
PA-5050		<ul style="list-style-type: none"> • 10 Gbps firewall throughput (App-ID enabled1) • 5 Gbps threat prevention throughput • 2,000,000 max sessions • 120,000 new sessions per second • 125 virtual routers • 25/125 virtual systems (base/max) • 500 security zones • 20,000 max number of policies
PA-5060		<ul style="list-style-type: none"> • 20 Gbps firewall throughput (App-ID enabled1) • 10 Gbps threat prevention throughput • 4,000,000 max sessions • 120,000 new sessions per second • 225 virtual routers • 25/225 virtual systems (base/max) • 900 security zones • 40,000 max number of policies
PA-5220		<ul style="list-style-type: none"> • 18.5 Gbps firewall throughput (App-ID enabled) • 9.2 Gbps threat prevention throughput • 4,000,000 max sessions • 169,000 new sessions per second • 20 virtual routers • 10/20 virtual systems (base/max)

Product Identification	Illustration	Description
PA-5250		<ul style="list-style-type: none"> • 35.9 Gbps firewall throughput (App-ID enabled) • 20.4 Gbps threat prevention throughput • 8,000,000 max sessions • 348,000 new sessions per second • 125 virtual routers • 25/125 virtual systems (base/max)
PA-5260		<ul style="list-style-type: none"> • 72.3 Gbps firewall throughput (App-ID enabled) • 30.2 Gbps threat prevention throughput • 32,000,000 max sessions • 458,000 new sessions per second • 225 virtual routers • 25/225 virtual systems (base/max)
PA-7050		<ul style="list-style-type: none"> • 120 Gbps Firewall throughput (App-ID enabled) • 100 Gbps Threat prevention throughput (DSRI Enabled2) • 60 Gbps Threat prevention throughput • 24,000,000 Max sessions • 720,000 New sessions per second • 25/225 Virtual systems (base/max)
PA-7080		<ul style="list-style-type: none"> • 200 Gbps Firewall throughput (App-ID enabled) • 160 Gbps Threat prevention throughput (DSRI Enabled2) • 100 Gbps Threat prevention throughput • 40,000,000 Max sessions • 1,200,000 New sessions per second • 25/225 Virtual systems (base/max)
Virtual Appliances		
VM-50		<ul style="list-style-type: none"> • 50,000 max sessions • 250 security rules • 1,000 dynamic IP addresses • 15 Security zones
VM-100		<ul style="list-style-type: none"> • 250,000 max sessions • 1,500 security rules • 2,500 dynamic IP addresses • 40 Security zones

Product Identification	Illustration	Description
VM-200		<ul style="list-style-type: none"> • 250,000 max sessions • 1,500 security rules • 2,500 dynamic IP addresses • 40 Security zones
VM-300		<ul style="list-style-type: none"> • 800,000 max sessions • 10,000 security rules • 100,000 dynamic IP addresses • 40 Security zones
VM-500		<ul style="list-style-type: none"> • 2,000,000 max sessions • 10,000 security rules • 100,000 dynamic IP addresses • 200 Security zones
VM-700		<ul style="list-style-type: none"> • 10,000,000 max sessions • 20,000 security rules • 100,000 dynamic IP addresses • 200 Security zones
VM-1000-HV		<ul style="list-style-type: none"> • 800,000 max sessions • 10,000 security rules • 100,000 dynamic IP addresses • 40 Security zones

2.4.2 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

2.4.2.1 Security audit

The TOE is designed to be able to generate logs for a wide range of security relevant events including the events specified in [NDcPP] and [FWcPP]. The TOE can be configured to store the logs locally so they can be accessed by an administrator and can also be configured to send the logs to a designated external log server.

2.4.2.2 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 and TLS. Note that to be in the evaluated configuration, the TOE must be configured in Common Criteria mode, which ensures the TOE's configuration is consistent with the FIPS 140-2 standard.

2.4.2.3 User data protection

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

2.4.2.4 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 (HTTP over TLS) and direct connections to the GUI for interactive administrator sessions.

The TOE supports the local (i.e., on device) definition and authentication of administrators with username, password, and role (set of privileges), which it uses to authenticate the human user and to associate that user with an authorized role. In addition, the TOE can authenticate users using X509 certificates and can be configured to lock a user out after a configurable number of unsuccessful authentication attempts.

2.4.2.5 Security management

The TOE provides a GUI to access the wide range of 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 locally via direct RJ-45 Ethernet cable connection and remotely using an HTTPS/TLS 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 create new user accounts, configure the audit function, configure the information flow control rules, and review the audit trail. The TOE provides pre-defined Security Administrator, Audit Administrator, and Cryptographic Administrator roles. These administrator roles are all considered Security Administrator as defined in the [NDcPP] for the purposes of this ST.

2.4.2.6 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 might detect when it is failing. It also includes mechanisms so that the TOE itself can be updated while ensuring that the updates will not introduce malicious or other unexpected changes in the TOE.

2.4.2.7 TOE access

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

2.4.2.8 Trusted path/channels

The TOE protects interactive communication with remote administrators using IPsec or HTTP over TLS. IPsec and TLS ensures both integrity and disclosure protection.

The TOE protects communication with the UIA and update server using TLS connections; the external log server with IPsec or TLS to prevent unintended disclosure or modification of the transferred data.

2.4.2.9 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..

2.5 TOE Documentation

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

For PAN-OS v8.0.6, these documents include:

- Palo Alto Networks Common Criteria Evaluated Configuration Guide (CCECG) for PAN-OS v8.0.6, Version 1.0, December 19, 2017
- Palo Alto Networks PAN-OS Administrator's Guide Version 8.0, February 3, 2017
- Palo Alto Networks PAN-OS Web Interface Reference Guide, Version 8.0, February 6, 2017
- Palo Alto Networks VM - Series Deployment Guide, Version 8.0, January 30, 2018

3. Security Problem Definition

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

In general, the [NDcPP] has presented a Security Problem Definition appropriate for network infrastructure devices, such as firewalls, and as such is applicable to the Palo Alto TOE. Likewise, the [FWcPP] has presented a Security Problem definition appropriate for Stateful Traffic Filter Firewalls, as such both are applicable to the Palo Alto TOE.

4. Security Objectives

Like the Security Problem Definition, this security target includes by reference the Security Objectives from the [NDcPP] and [FWcPP]. 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] and [FWcPP] have presented Security Objectives appropriate for network infrastructure devices, such as firewalls, Stateful Traffic Filter Firewalls, as such are applicable to the Palo Alto TOE.

4.1 Security Objectives for the Operational Environment

OE.NO_GENERAL_PURPOSE	There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE.
OE.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.
OE.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	TOE Administrators are trusted to follow and apply all guidance documentation in a trusted manner.

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 following Protection Profiles (PP):

- *collaborative Protection Profile for Network Devices, Version 1.0, 27 February 2015* [NDcPP],
- *collaborative Protection Profile for Stateful Traffic Filter Firewalls, Version 1.0, 27 February 2015* [FWcPP].

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

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

5.1 Extended Requirements

All of the extended requirements in this ST have been drawn from the [NDcPP] and [FWcPP]. The [NDcPP], and [FWcPP] define the following extended SFRs and since they are not redefined in this ST, the [NDcPP] and [FWcPP] should be consulted for more information in regard to those CC extensions.

- FAU_STG_EXT.1: Protected Audit Event Storage
- FAU_STG_EXT.3: Display warning for local storage space
- FCS_HTTPS_EXT.1: HTTPS Protocol
- FCS_IPSEC_EXT.1: IPsec Protocol
- FCS_RBG_EXT.1: Random Bit Generation
- FCS_TLSC_EXT.1 - TLS Client Protocol
- FCS_TLSC_EXT.2 - TLS Client Protocol with authentication
- FCS_TLSS_EXT.1 - TLS Server Protocol
- FCS_TLSS_EXT.2 - TLS Server Protocol with mutual authentication
- FFW_RUL_EXT.1 Stateful Traffic Filtering
- FFW_RUL_EXT.2 Stateful Filtering of Dynamic Protocols
- FIA_PMG_EXT.1: Password Management
- FIA_UAU_EXT.2: Extended: Password-based Authentication Mechanism
- FIA_UIA_EXT.1: User Identification and Authentication
- FIA_X509_EXT.1 – X.509 Certificate Validation
- FIA_X509_EXT.2 – X.509 Certificate Authentication
- FIA_X509_EXT.3 – X.509 Certificate Requests
- FPT_APW_EXT.1: Extended: Protection of Administrator Passwords
- FPT_SKP_EXT.1: Extended: Protection of TSF Data (for reading of all symmetric keys)
- FPT_TST_EXT.1: TSF Testing
- FPT_TST_EXT.2: Extended: TSF testing

- FPT_TUD_EXT.1: Extended: Trusted Update
- FTA_SSL_EXT.1: TSF-initiated Session Locking
- FTP_ITC_EXT.1.1: Inter-TSF Trusted Channel

5.2 TOE Security Functional Requirements

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

Table 2 TOE Security Functional Components

Requirement Class	Requirement Component
FAU: Security audit	FAU_GEN.1: Audit Data Generation
	FAU_GEN.2: User identity association
	FAU_STG.1: Protected audit trail storage
	FAU_STG_EXT.3: Display warning for local storage space
	FAU_STG_EXT.1: Protected Audit Event Storage
FCS: Cryptographic support	FCS_CKM.1: Cryptographic Key Generation
	FCS_CKM.2: Cryptographic Key Establishment
	FCS_CKM.4 Cryptographic Key Destruction
	FCS_COP.1(1): Cryptographic Operation (AES Data Encryption/Decryption)
	FCS_COP.1(2) Cryptographic Operation (Signature Generation and Verification)
	FCS_COP.1(3) Cryptographic Operation (Hash Algorithm)
	FCS_COP.1(4) Cryptographic Operation (Keyed Hash Algorithm)
	FCS_HTTPS_EXT.1: HTTPS Protocol
	FCS_IPSEC_EXT.1: IPsec Protocol
	FCS_RBG_EXT.1: Random Bit Generation
	FCS_TLSC_EXT.1 - TLS Client Protocol
	FCS_TLSC_EXT.2 - TLS Client Protocol with authentication
	FCS_TLSS_EXT.1 - TLS Server Protocol
	FCS_TLSS_EXT.2 - TLS Server Protocol with mutual authentication
FDP: User data protection	FDP_RIP.2: Full Residual Information Protection
FIA: Identification and authentication	FIA_PMG_EXT.1: Password Management
	FIA_UAU.7: Protected Authentication Feedback
	FIA_UAU_EXT.2: Extended: Password-based Authentication Mechanism
	FIA_UIA_EXT.1: User Identification and Authentication
	FIA_X509_EXT.1 Extended: X.509 Certificates
	FIA_X509_EXT.2: X.509 Certificate Authentication
	FIA_X509_EXT.3: X.509 Certificate Requests
FFW: Stateful Traffic Filtering	FFW_RUL_EXT.1: Stateful Traffic Filtering
	FFW_RUL_EXT.2: Stateful Filtering of Dynamic Protocols
FMT: Security management	FMT_MOF.1(1)/TrustedUpdate: Management of security functions behaviour
	FMT_MOF.1(2)/TrustedUpdate: Management of security functions behaviour

Requirement Class	Requirement Component
	FMT_MOF.1(1)/Audit: Management of security functions behaviour
	FMT_MTD.1: Management of TSF Data (for general TSF data)
	FMT_MTD.1/AdminAct: Management of TSF Data
	FMT_SMF.1: Specification of Management Functions
	FMT_SMR.2: Restrictions on Security Roles
FPT: Protection of the TSF	FPT_APW_EXT.1: Protection of Administrator Passwords
	FPT_SKP_EXT.1: Protection of TSF Data (for reading of all symmetric keys)
	FPT_STM.1: Reliable Time Stamps
	FPT_TST_EXT.1: TSF Testing
	FPT_TST_EXT.2: Extended: TSF Testing
	FPT_TUD_EXT.1: Extended: Trusted Update
FTA: TOE access	FTA_SSL.3: TSF-initiated Termination
	FTA_SSL.4: User-initiated Termination
	FTA_SSL_EXT.1: TSF-initiated Session Locking
	FTA_TAB.1: Default TOE Access Banners
FTP: Trusted path/channels	FTP_ITC.1: Inter-TSF trusted channel
	FTP_TRP.1: Trusted Path

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 user account shall be logged if individual user accounts are required for administrators).
 - Security related 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).
 - Starting and stopping services (if applicable)
 - **[no other actions]**;
- d) Specifically defined auditable events listed in **Table 3**.

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

- a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event; and
- b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, information specified in column three of **Table 3**.

Table 3 Auditable Events

Requirement	Auditable Events	Additional Audit Record Contents
FAU_GEN.1	None.	None.
FAU_GEN.2	None.	None.
FAU_STG.1	None	None.
FAU_STG_EXT.1	None.	None.
FAU_STG_EXT.3	Warning about low storage space for audit events.	None
FCS_CKM.1	None.	None.
FCS_CKM.1.1/IKE	None.	None.
FCS_CKM.2	None.	None.
FCS_CKM.4	None.	None.
FCS_COP.1(1)	None.	None.
FCS_COP.1(2)	None.	None.
FCS_COP.1(3)	None.	None.
FCS_COP.1(4)	None.	None.
FCS_HTTPS_EXT.1	Failure to establish an HTTPS session.	Reason for failure
FCS_IPSEC_EXT.1	Failure to establish an IPsec SA.	Reason for failure.
FCS_RBG_EXT.1	None.	None
FCS_TLSC_EXT.1	Failure to establish a TLS session.	Reason for failure
FCS_TLSC_EXT.2	Failure to establish a TLS session.	Reason for failure
FCS_TLSS_EXT.1	Failure to establish a TLS session.	Reason for failure
FCS_TLSS_EXT.2	Failure to establish a TLS session.	Reason for failure
FDP_RIP.2	None.	None.
FIA_PMG_EXT.1	None.	None.
FIA_UAU_EXT.2	All use of identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UIA_EXT.1	All use of identification and authentication mechanism.	Provided user identity, origin of the attempt (e.g., IP address).
FIA_UAU.7	None.	None.
FIA_X509_EXT.1	Unsuccessful attempt to validate a certificate	Reason for failure
FIA_X509_EXT.2	None.	None.
FIA_X509_EXT.3	None.	None.
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

Requirement	Auditable Events	Additional Audit Record Contents
	Indication of packets dropped due to too much network traffic	TOE interface that is unable to process packets Identifier of rule causing packet drop
FFW_RUL_EXT.2	The App-ID has been used with the protocol processed	Protocol name
FMT_MOF.1(1)/AdminAct	Modification of the behaviour of the TSF.	None.
FMT_MOF.1(2)/AdminAct	Starting and stopping of services.	None.
FMT_MOF.1(1)/Audit	Modification of the behaviour of the transmission of audit data to an external IT entity.	None.
FMT_MOF.1(1)/TrustedUpdate	Any attempt to initiate a manual update.	None.
FMT_MOF.1(2)/TrustedUpdate	Enabling or Disabling automatic checking for updates or automatic updates.	None.
FMT_MTD.1	All management activities of TSF data.	None.
FMT_MTD.1/AdminAct	Modification, deletion, generation/import of cryptographic keys.	None.
FMT_SMF.1	None.	None.
FMT_SMR.2	None.	None.
FPT_APW_EXT.1	None.	None.
FPT_SKP_EXT.1	None.	None.
FPT_STM.1	Changes to the time.	The old and new values for the time. Origin of the attempt to change time for success and failure (e.g., IP address).
FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure)	No additional information.
FPT_TST_EXT.1	None.	None.
FTA_SSL_EXT.1	Any attempts at unlocking of an interactive session.	None.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	None.
FTA_SSL.4	The termination of an interactive session.	None.
FTA_TAB.1	None.	None.

Requirement	Auditable Events	Additional Audit Record Contents
FTP_ITC.1	Initiation of the trusted channel. Termination of the trusted channel. Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempt.
FTP_TRP.1	Initiation of the trusted path. Termination of the trusted path. Failures of the trusted path functions.	Identification of the claimed user identity.

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.1 - Protected audit trail storage

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

FAU_STG.1.2 The TSF shall be able to prevent unauthorised modifications to the stored audit records in the audit trail.

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.

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

FAU_STG_EXT.1.3 The TSF shall [*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.3 - Display warning for local storage space

FAU_STG_EXT.3.1 The TSF shall generate a warning to inform the user before the local space to store audit data is used up and/or the TOE will lose audit data due to insufficient local space.

5.2.2 Cryptographic Support (FCS)

FCS_CKM.1 – Cryptographic Key Generation

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

- *RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.3;*
- *ECC schemes using “NIST curves” [P-256, P-384, P-521] that meet the following: FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.4;*

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

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

FCS_CKM.2 – Cryptographic Key Establishment

FCS_CKM.2.1¹

The TSF shall perform cryptographic key establishment in accordance with a specified cryptographic key establishment method: [

- *RSA-based key establishment schemes that meets the following: NIST Special Publication 800-56B Revision 1, “Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography”;*
- *Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 2, “Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography”;*
- *Finite field-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 2, “Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography*

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

FCS_CKM.4 – Cryptographic Key Destruction²

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 [*selection:*
 - *logically addresses the storage location of the key and performs a [selection: single, [assignment: three or more times] overwrite consisting of [selection: a pseudo-random pattern using the TSF’s RBG, zeroes, ones, a new value of the key, [assignment: using a different alternating pattern]]];*
 - *instructs a part of the TSF to destroy the abstraction that represents the key]]*

that meets the following: No Standard.

Application Note: The TOE does not store plain text keys in non-volatile storage. NIAP TRRT 241 response stated: “The TRRT does not see the need to modify the requirement. If the TOE does not store plaintext keys in one type of memory, that portion of the requirement is met. A statement in the TSS that plaintext keys are not stored in a specific type of memory is sufficient.”

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

FCS_COP.1.1(1)

The TSF shall perform encryption/decryption in accordance with a specified cryptographic algorithm AES used in [*GCM, CBC*] mode and cryptographic key sizes [*128 bits, 256 bits*]

¹ Updated per NIAP Technical Decision TD0235

² FCS_CKM.4 has been modified to comply with TD0130. Please see Section 7 Table 6 for more details.

that meet the following: AES as specified in ISO 18033-3, [*CBC as specified in ISO 10116, GCM as specified in ISO 19772*].

FCS_COP.1(2) – Cryptographic Operation (Signature Generation and Verification)

FCS_COP.1.1(2)³ The TSF shall perform cryptographic signature services (generation and verification) in accordance with a specified cryptographic algorithm [

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

]

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(3) – Cryptographic Operation (Hash Algorithm)

FCS_COP.1.1(3) 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] that meet the following: ISO/IEC 10118-3:2004.

FCS_COP.1(4) – Cryptographic Operation (Keyed Hash Algorithm)

FCS_COP.1.1(4) The TSF shall perform keyed-hash message authentication in accordance with a specified cryptographic algorithm [*HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512*] and cryptographic key sizes [256, 448, 512, 1536, 2048 bits for HMAC-SHA-1 and HMAC-SHA-256; 256, 448, 1024, 1536, and 2048 bits for HMAC-SHA-384 and HMAC-SHA-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_HTTPS_EXT.1 – HTTPS Protocol

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

FCS_HTTPS_EXT.1.2 The TSF shall implement HTTPS using TLS.

FCS_HTTPS_EXT.1.3⁴ The TSF shall establish the connection only if [*the peer presents a valid certificate during handshake, the peer initiates handshake*].

³ FCS_COP.1.1(2) has been modified to comply with TD0116 and TD0199.

⁴ Updated per NIAP Technical Decision TD0125

FCS_IPSEC_EXT.1 – IPsec Protocol

- FCS_IPSEC_EXT.1.1** The TSF shall implement the IPsec architecture as specified in RFC 4301.
- FCS_IPSEC_EXT.1.2** The TSF shall have a nominal, final entry in the SPD that matches anything that is otherwise unmatched, and discards it.
- FCS_IPSEC_EXT.1.3⁵** The TSF shall implement [*tunnel mode*].
- FCS_IPSEC_EXT.1.4⁶** The TSF shall implement the IPsec protocol ESP as defined by RFC 4303 using the cryptographic algorithms [*AES-CBC-128 (specified by RFC 3602), AES-CBC-256 (specified by RFC 3602), AES-GCM-128 (specified in RFC 4106), AES-GCM-256 (specified in RFC 4106)*] together with a Secure Hash Algorithm (SHA)-based HMAC.
- FCS_IPSEC_EXT.1.5** The TSF shall implement the protocol: [
 - IKEv1, using Main Mode for Phase 1 exchanges, as defined in RFCs 2407, 2408, 2409, RFC 4109, [*RFC 4304 for extended sequence numbers*], and [*no other RFCs for hash functions*];
 - IKEv2 as defined in RFC 5996 and [*with mandatory support for NAT traversal as specified in RFC 5996, section 2.23*], and [*RFC 4868 for hash functions*]].
- FCS_IPSEC_EXT.1.6⁷** The TSF shall ensure the encrypted payload in the [*IKEv1, IKEv2*] protocol uses the cryptographic algorithms [*AES-CBC-128 (as specified in RFC 3602), AES-CBC-256 (as specified in RFC 3602)*].
- FCS_IPSEC_EXT.1.7** The TSF shall ensure that [
 - *IKEv1 Phase 1 SA lifetimes can be configured by an Security Administrator based on*
[
 - *length of time, where the time values can configured within [3 minutes to 8760] hours;*];
 - *IKEv2 SA lifetimes can be configured by an Security Administrator based on*
[
 - *length of time, where the time values can configured within [3 minutes to 8760] hours*]].].
- FCS_IPSEC_EXT.1.8** The TSF shall ensure that [
 - *IKEv1 Phase 2 SA lifetimes can be configured by a Security Administrator based on*
[
 - *number of bytes;*
 - *length of time, where the time values can be configured within [3 minutes to 8760] hours;*];
 - *IKEv2 Child SA lifetimes can be configured by a Security Administrator based on*
[
 - *number of bytes;*]].

⁵ Updated per NIAP Technical Decision TD0160

⁶ Updated per NIAP Technical Decision TD0225

⁷ Updated per NIAP Technical Decision TD0225

- *length of time, where the time values can be configured within [3 minutes 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 $g^x \text{ mod } p$) using the random bit generator specified in FCS_RBG_EXT.1, and having a length of at least [224 (for DH Group 14), 256 (for DH Group 19), 384 (for DH Group 20)] bits.
- FCS_IPSEC_EXT.1.10** The TSF shall generate nonces used in [IKEv1, IKEv2] exchanges of length [
- *At least 128 bits in size and at least half the output size of the negotiated pseudorandom function (PRF) hash*
-].
- FCS_IPSEC_EXT.1.11**⁸ The TSF shall ensure that IKE protocols implement DH Group(s) [14 (2048-bit MODP), 19 (256-bit Random ECP), 20 (384-bit Random ECP)].
- FCS_IPSEC_EXT.1.12** The TSF shall be able to ensure by default that the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the [IKEv1 Phase 1, IKEv2 IKE_SA] connection is greater than or equal to the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the [IKEv1 Phase 2, IKEv2 CHILD_SA] connection.
- FCS_IPSEC_EXT.1.13** The TSF shall ensure that all IKE protocols perform peer authentication using a [RSA, ECDSA] that use X.509v3 certificates that conform to RFC 4945 and [no other method].
- FCS_IPSEC_EXT.1.14**⁹ 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 types: [IP address, Fully Qualified Domain Name (FQDN), user FQDN, Distinguished Name (DN)] and [no other reference identifier type]].

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 software-based noise source, one hardware-based noise source] with minimum of [256 bits] of entropy at least equal to the greatest security strength, according to ISO/IEC 18031:2011 Table C.1 “Security Strength Table for Hash Functions”, of the keys and hashes that it will generate.

FCS_TLSC_EXT.1 - TLS Client Protocol

- FCS_TLSC_EXT.1.1**¹⁰ The TSF shall implement [TLS 1.2 (RFC 5246), TLS 1.1 (RFC 4346)] supporting the following ciphersuites: [
- *TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*
 - *TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*
 - *TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*
 - *TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*
 - *TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492*

⁸ Updated per NIAP Technical Decision TD0224

⁹ Updated per NIAP Technical Decision TD0223

¹⁰ Updated per NIAP Technical Decision TD0226

- *TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492*
- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492*
- *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492*
- *TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246*
- *TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246*
- *TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246*
- *TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246*
- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*

].

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

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

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

FCS_TLSC_EXT.2 - TLS Client Protocol with authentication

FCS_TLSC_EXT.2.1¹¹ The TSF shall implement [*TLS 1.2 (RFC 5246), TLS 1.1 (RFC 4346)*] supporting the following ciphersuites: [

- *TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*
- *TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*
- *TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*
- *TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*
- *TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492*
- *TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492*
- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492*
- *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492*
- *TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246*
- *TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246*
- *TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246*
- *TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246*
- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*

].

FCS_TLSC_EXT.2.2 The TSF shall verify that the presented identifier matches the reference identifier according to RFC 6125.

FCS_TLSC_EXT.2.3 The TSF shall only establish a trusted channel if the peer certificate is valid.

¹¹ Updated per NIAP Technical Decision TD0226

- FCS_TLSC_EXT.2.4** The TSF shall present the Supported Elliptic Curves Extension in the Client Hello with the following NIST curves: [*secp256r1*, *secp384r1*, *secp521r1*] and no other curves.
- FCS_TLSC_EXT.2.5** The TSF shall support mutual authentication using X.509v3 certificates.

FCS_TLSS_EXT.1 - TLS Server Protocol

- FCS_TLSS_EXT.1.1¹²** The TSF shall implement [*TLS 1.2 (RFC 5246)*, *TLS 1.1 (RFC 4346)*] supporting the following ciphersuites: [
- *TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*
 - *TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*
 - *TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246*
 - *TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246*
 - *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492*
 - *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492*
 - *TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
 - *TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
-].
- FCS_TLSS_EXT.1.2¹³** The TSF shall deny connections from clients requesting SSL 2.0, SSL 3.0, TLS 1.0, and [*none*].
- FCS_TLSS_EXT.1.3¹⁴** The TSF shall [*perform RSA key establishment with key size [2048 bits, 3072 bits], generate EC Diffie-Hellman parameters over NIST curves [secp256r1, secp384r1, secp521r1] and no other curves; generate Diffie-Hellman parameters of size [2048 bits]*].

FCS_TLSS_EXT.2 - TLS Server Protocol with mutual authentication

- FCS_TLSS_EXT.2.1¹⁵** The TSF shall implement [*TLS 1.2 (RFC 5246)*, *TLS 1.1 (RFC 4346)*] supporting the following ciphersuites: [
- *TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268*
 - *TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268*
 - *TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246*
 - *TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246*
 - *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492*
 - *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492*
 - *TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
 - *TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
-].
- FCS_TLSS_EXT.2.2¹⁶** The TSF shall deny connections from clients requesting SSL 2.0, SSL 3.0, TLS 1.0, and [*none*].
- FCS_TLSS_EXT.2.3¹⁷** The TSF shall [*perform RSA key establishment with key size [2048 bits, 3072 bits], generate EC Diffie-Hellman parameters over NIST curves [secp256r1, secp384r1, secp521r1] and no other curves; generate Diffie-Hellman parameters of size [2048, bits]*].

¹² Updated per NIAP Technical Decision TD0226

¹³ Updated per NIAP Technical Decision TD0156

¹⁴ Updated per NIAP Technical Decision TD0226

¹⁵ Updated per NIAP Technical Decision TD0226

¹⁶ Updated per NIAP Technical Decision TD0156

¹⁷ Updated per NIAP Technical Decision TD0226

- FCS_TLSS_EXT.2.4** The TSF shall support mutual authentication of TLS clients using X.509v3 certificates.
- FCS_TLSS_EXT.2.5** The TSF shall not establish a trusted channel if the peer certificate is invalid.
- FCS_TLSS_EXT.2.6** The TSF shall not establish a trusted channel if the distinguished name (DN) or Subject Alternative Name (SAN) contained in a certificate does not match the expected identifier for the peer.

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_PMG_EXT.1 – Password management

- FIA_PMG_EXT.1.1** The TSF shall provide the following password management capabilities for administrative passwords:
1. Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters: [“!”, “@”, “#”, “\$”, “%”, “^”, “&”, “*”, “(”, “)”, [“”, “+”, “,”, “-”, “.”, “/”, “:”, “;”, “<”, “=”, “>”, “[”, “\”, “]”, “_”, “`”, “{”, “}”, and “~”];
 2. Minimum password length shall be settable by the Security Administrator, and support passwords of 15 characters or greater.

FIA_UAU.7 – Protected authentication feedback

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

FIA_UAU_EXT.2 – Extended: Password-based authentication mechanism

- FIA_UAU_EXT.2.1** The TSF shall provide a local password-based authentication mechanism, [*X509 certificates*] to perform administrative user authentication.

FIA_UIA_EXT.1 – User identification and authentication

- FIA_UIA_EXT.1.1** The TSF shall allow the following actions prior to requiring the non-TOE entity to initiate the identification and authentication process:
- Display the warning banner in accordance with FTA_TAB.1;
 - [*no other actions*].
- FIA_UIA_EXT.1.2** The TSF shall require each administrative user to be successfully identified and authenticated before allowing any other TSF-mediated actions on behalf of that administrative user.

FIA_X509_EXT.1 – X.509 Certificate Validation

- FIA_X509_EXT.1.1¹⁸** The TSF shall validate certificates in accordance with the following rules:
- RFC 5280 certificate validation and certificate path validation.

¹⁸ Updated per NIAP Technical Decision TD0169

- The certificate path must terminate with a trusted CA certificate.
- The TSF shall validate a certificate path by ensuring the presence of the basicConstraints extension and that the CA flag is set to TRUE for all CA certificates.
- The TSF shall validate the revocation status of the certificate using [*the Online Certificate Status Protocol (OCSP) as specified in RFC 2560, 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 TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.
 - Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.
 - OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.

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

FIA_X509_EXT.2 – X.509 Certificate Authentication

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

FIA_X509_EXT.2.2 When the TSF cannot establish a connection to determine the validity of a certificate, the TSF shall [*allow the administrator to choose whether to accept the certificate in these cases*].

FIA_X509_EXT.3 – X.509 Certificate Requests

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

5.2.5 Stateful Traffic Filtering (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).**
 - **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.*
 - *drop all inbound and outbound IPv6 packets containing a Hop-by-Hop header with option type values intended for Destination Options.*
 - *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]*.

5.2.6 Security Management (FMT)

FMT_MOF.1(1)/TrustedUpdate - Management of Security Functions Behaviour Functions

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

FMT_MOF.1(2)/TrustedUpdate – Management of Security Functions Behavior

- FMT_MOF.1(2)/TrustedUpdate** The TSF shall restrict the ability to enable, disable the functions *[automatic checking for updates, automatic update]* to Security Administrators.

FMT_MOF.1(1)/AdminAct – Management of Security Functions Behavior

- FMT_MOF.1.1(1)/AdminAct** The TSF shall restrict the ability to modify the behaviour of the functions TOE Security Functions to Security Administrators.

FMT_MOF.1(2)/AdminAct – Management of Security Functions Behavior

FMT_MOF.1.1(2)/AdminAct The TSF shall restrict the ability to enable, disable the functions services to Security Administrators..

FMT_MOF.1(1)/Audit - Management of security functions behavior

FMT_MOF.1.1(1)/Audit The TSF shall restrict the ability to determine the behaviour of, modify the behaviour of the functions transmission of audit data to an external IT entity to Security Administrators.

FMT_MTD.1 – Management of TSF Data

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

FMT_MTD.1/AdminAct - Management of TSF Data

FMT_MTD.1.1/AdminAct The TSF shall restrict the ability to modify, delete, generate/import the cryptographic keys to Security Administrators.

FMT_SMF.1 – Specification of management functions

FMT_SMF.1.1¹⁹ The TSF shall be capable of performing the following management functions:

- Ability to administer the TOE locally and remotely;
- Ability to configure the access banner;
- Ability to configure the session inactivity time before session termination or locking;
- Ability to update the TOE, and to verify the updates using [*digital signature*] capability prior to installing those updates;
- Ability to configure firewall rules;
- [
- *Ability to configure audit behavior;*
- *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;*
-].

FMT_SMR.2 – Restrictions on security roles

FMT_SMR.2.1 The TSF shall maintain the roles:

- Security Administrator.

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

FMT_SMR.2.3 The TSF shall ensure that the conditions

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

¹⁹ This requirement was modified per TD0090

5.2.7 Protection of the TSF (FPT)

FPT_APW_EXT.1 – Protection of administrator passwords

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

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

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

FPT_STM.1 – Reliable time stamps

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

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), periodically during normal operation, at the request of the authorized user*] to demonstrate the correct operation of the TSF: [

- AES Encrypt Known Answer Test
- AES Decrypt Known Answer Test
- AES GCM Encrypt Known Answer Test
- AES GCM Decrypt Known Answer Test
- AES CCM Encrypt Known Answer Test
- AES CCM Decrypt Known Answer Test
- RSA Sign Known Answer Test
- RSA Verify Known Answer Test
- ECDSA Sign Known Answer Test
- ECDSA Verify Known Answer Test
- HMAC-SHA-1 Known Answer Test
- HMAC-SHA-256 Known Answer Test
- HMAC-SHA-384 Known Answer Test
- HMAC-SHA-512 Known Answer Test
- SHA-1 Known Answer Test
- SHA-256 Known Answer Test
- SHA-384 Known Answer Test
- SHA-512 Known Answer Test
- DRBG SP800-90A Known Answer Tests
- SP 800-90A Section 11.3 Health Tests
- DH Known Answer Test
- ECDH Known Answer Test
- Firmware Integrity Test –verified with HMAC-SHA-256 and ECDSA P-256. If the calculated result does not equal the previously generated result, the software/firmware test shall fail.

]

FPT_TST_EXT.2 – Extended: TSF testing

- FPT_TST_EXT.2.1** The TSF shall fail self-testing if a certificate is used for self tests and the corresponding certificate is deemed invalid.

FPT_TUD_EXT.1 – Extended: Trusted update

- FPT_TUD_EXT.1.1²⁰** The TSF shall provide Security Administrators the ability to query the currently executing version of the TOE firmware/software and [*the most recently installed version of the TOE firmware/software*].
- FPT_TUD_EXT.1.2** The TSF shall provide Security Administrators the ability to manually initiate updates to TOE firmware/software and [*no other update mechanism*].
- FPT_TUD_EXT.1.3** The TSF shall provide a means to authenticate firmware/software updates to the TOE using a [*digital signature mechanism*] prior to installing those updates.

5.2.8 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 Administrator-initiated termination of the Administrator’s own interactive session.

FTA_SSL_EXT.1 – TSF-initiated session locking

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

FTA_TAB.1 – Default TOE access banners

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

5.2.9 Trusted Path/Channels (FTP)

FTP_ITC.1 – Inter-TSF Trusted Channel

- FTP_ITC.1.1** The TSF shall be capable of using [*IPsec, TLS*] to provide a trusted communication channel between itself and authorized IT entities supporting the following capabilities: audit server, [*update server, connections with UIA, connections to WildFire, connections to Panorama*] that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from disclosure and detection of modification of the channel data.
- FTP_ITC.1.2** The TSF shall permit the TSF, or the authorized IT entities to initiate communication via the trusted channel.
- FTP_ITC.1.3** The TSF shall initiate communication via the trusted channel for [
 - **transmitting audit records to an audit server using IPsec or TLS,**
 - **to retrieve the IP address mapping information with UIA using TLS,**
 - **receiving TOE updates from the update server using TLS,**

²⁰ Updated per NIAP Technical Decision TD0154

- **communicating to WildFire and Panorama Management System using TLS].**

FTP_TRP.1 – Trusted path

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

5.3 TOE Security Assurance Requirements

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

Table 4 Assurance Components

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

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

- Supporting Document Mandatory Technical Document: Evaluation Activities for Network Device cPP, February-2015, Version 1.0
- Supporting Document Mandatory Technical Document Evaluation Activities for Stateful Traffic Filter Firewalls cPP, February-2015, Version 1.0

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

6.1 Security Audit

The TOE is designed to be able to generate log records for a wide range of 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, any use of an administrator command via the Web Interface, as well as all of the events identified in **Table 3** (which corresponds to the audit events specified in the [NDcPP] and [FWcPP]).

The logged audit records identify the date and time, the nature or type of the triggering event, an indication of whether the event succeeded or failed, and the identity of the user responsible for the event. The name of the relevant key is recorded in the audit record whenever an administrator generates, imports, changes, or deletes a cryptographic key. The logged audit records also include event-specific content that includes at least all of the content required in **Table 3**.

The audit trail generated by the TOE comprises several logs, which are locally stored in the PAN-OS file system on the hard disk:

- Configuration logs—include events such as when an administrator configures the security policies, and when an administrator configures which events gets audited
- System logs—record user login and logout
- 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 from the Web Interface by specifying the percentage of space allocated to each log type on the hard disk. If the log size is reduced, the firewall removes the oldest logs when the changes are committed. When a log reaches the maximum size, the firewall 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. For example, for a 120GB drive approximately 83GB is allocated for logging. Platform capabilities range from a limit of 3-4GB for the PA-200 which has a 16GB flash drive and up to a maximum of 4TBs for the larger PA-7000 Series platforms.

The user is warned before local storage for audit data is full. The threshold for issuing the warning is specified by the administrator in terms of % full of the various log files comprising the audit trail. The TOE generates an alarm that is displayed in a window on the web GUI of a logged-in administrator. If no administrator sessions are active when the alarm is generated, it can subsequently be viewed by clicking on the Alarms icon at the bottom of the GUI. All alarms remain available for display until acknowledged. The TOE also generates an audit record when the storage threshold is reached.

The TOE stores the audit records locally and protects them from unauthorized deletion by allowing only users in the pre-defined Audit Administrator role to access the audit trail with delete privileges. The pre-defined Audit Administrator role is part of the Security Administrator role as defined by the [NDcPP]. The TOE does not provide an interface where a user can modify the audit records, thus it prevents modification to the audit records.

The TOE can be configured to send generated audit records to an external Syslog server 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.

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

- FAU_GEN.1—the TOE can generate audit records for events include starting and stopping the audit function, administrator commands, and all other events identified in **Table 3**. Furthermore, each audit record identifies the date/time, event type, outcome of the event, responsible subject/user, as well as the additional event-specific content indicated in **Table 3**.
- FAU_GEN.2—the TOE identifies the responsible user for each event based on the specific administrator or network entity (identified by IP address) that caused the event.
- FAU_STG.1-The amount of audit stored locally is by specified by the percentage of space allocated to each log type on the hard disk. The TOE stores the audit records locally and protects them from unauthorized deletion by allowing only users in the pre-defined Audit Administrator role to access the audit trail with delete privileges.
- FAU_STG_EXT.1—the TOE can be configured to export audit records to an external Syslog server and can be configured to use TLS or IPsec for communication with the Syslog server.
- FAU_STG_EXT.3- the TSF generates an alarm and an audit record to inform the user before the local space to store audit data is used up and/or the TOE will lose audit data due to insufficient local space.

6.2 Cryptographic Support

The TOE includes NIST-validated cryptographic algorithms providing supporting cryptographic functions. The following functions have been certified in accordance with the identified standards.

Table 5 Cryptographic Functions

Functions	Standards	Certificates
FCS_CKM.1 Asymmetric key generation		
FFC key pair generation (key size 2048 bits)	FIPS PUB 186-4	Appliances: DSA #1207 VMs: DSA #1205
ECC key pair generation (NIST curves P-256, P-384, P-521)	FIPS PUB 186-4	Appliances: ECDSA #1103 VMs: ECDSA #1101
RSA key generation (key size 2048, 3072 bits)	FIPS PUB 186-4	Appliances: RSA #2467 VMs: RSA #2463
FCS_CKM.2 Cryptographic Key Establishment		
ECDSA based key establishment	NIST SP 800-56A Revision 2	Appliances: CVL #1211 CVL # 1214 VMs: CVL #1206, CVL #1203
FFC based key establishment	NIST SP 800-56A Revision 2	Appliances: CVL 1211 VMs: # #1203
FCS_COP.1(1) AES Data Encryption/Decryption		
AES CBC, GCM (128, 256 bits)	AES as specified in ISO 18033-3 CBC as specified in ISO 10116 GCM as specified in ISO 19772	Appliances: AES # 4532 VMs: AES # 4526

Functions	Standards	Certificates
FCS_COP.1(2) Signature Generation and Verification		
RSA Digital Signature Algorithm (rDSA) (modulus 2048)	FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSAPKCS1v1_5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3	Appliances: RSA #2467 VMs: RSA #2463
ECDSA (NIST curves P-256, P-384, and P-521)	FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Section 6 and Appendix D, Implementing “NIST curves” P-256, P-384, ISO/IEC 14888-3, Section 6.4	Appliances: ECDSA # ECDSA 1103 VMs: # ECDSA 1101
FCS_COP.1(3) Cryptographic hashing		
SHA-1, SHA-256, SHA-384 and SHA-512 (digest sizes 160, 256, 384 and 512 bits)	ISO/IEC 10118-3:2004	Appliances: SHS #3713 VMs: SHS 3707
FCS_COP.1(4) Keyed-hash message authentication		
<ul style="list-style-type: none"> • HMAC-SHA-1 • HMAC-SHA-256 • HMAC-SHA-384 • HMAC-SHA-512 	ISO/IEC 9797-2:2011	Appliances: HMAC # 2990 VMs: HMAC #2986
FCS_RBG_EXT.1 Random bit generation		
CTR_DRBG (AES) from a hardware based noise source with one independent software-based noise source of 256 bits of non-determinism	ISO/IEC 18031:2011	Appliances: DRBG #1489 VMs: DRBG #1486

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 bit seed from the underlying Linux kernel pseudo-random number generator (PRNG). Entropy inputs are injected into the PRNG for initialization and through an updating mechanism. Entropy inputs are derived from the timing of IRQ event-driven interrupts (e.g., disk I/O completion events) and from a hardware-based noise source. On Palo Alto network devices, the noise source is a Cavium Octeon CPU, which is assumed to provide a full 256 bits of entropy per 256 random bits. On VM appliances, the noise source is the RDRAND/RDSEED instruction available on Intel Ivy Bridge architecture CPUs, which is assumed to provide 128 bits of entropy per 256 bits. The TOE generates asymmetric cryptographic keys used for key establishment in accordance with FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.3 for RSA schemes, FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.4 for ECC schemes and FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.1 for FFC schemes. The TOE generates asymmetric cryptographic keys used for signature generation and verification in accordance with FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.3 for RSA schemes.

While the TOE generally fulfills all of the FIPS PUB 186-4 requirements without extensions, the following table specifically identifies the “should”, “should not”, and “shall not” conditions from the publication along with an

indication of whether the TOE conforms to those conditions with deviations rationalized. Key generation is among the identified sections.

Table 6 FIPS 186-4 Conformance

FIPS PUB 186-4	“should”, “should not”, or “shall not”	Implemented accordingly?	Rationale for deviation
FIPS PUB 186-4 Appendix B.1			
B.1.1	should	Yes	N/A
B.1.2	should	Yes	N/A
FIPS PUB 186-4 Appendix B.3			
B.3.1	shall not	Yes	N/A
FIPS PUB 186-4 Appendix B.4			
B.4.1	should	Yes	N/A
B.4.2	should	Yes	N/A

The TOE performs cryptographic RSA-based key establishment in accordance with NIST Special Publication 800-56B, NIST Special Publication 800-56A for elliptic curve-based key establishment schemes, and NIST Special Publication 800-56A for finite field-based key establishment schemes. The TOE acts as both a sender and as a recipient for RSA-based key establishment schemes.

Table 7 Private Keys and CSPs

CSP #	CSP/Key Name	Type	Description
1	RSA Private Keys	RSA	RSA Private keys for verification of signatures, authentication or key establishment. (RSA 2048 or 3072-bit)
2	ECDSA Private Keys	ECDSA	ECDSA Private key for verification of signatures and authentication (P-256, P-384, P-521)
3	TLS PreMaster Secret	TLS Secret	Secret value used to derive the TLS session keys
4	TLS DH Private Components	DH	Diffie-Hellman private FFC or EC component used in TLS (DHE 2048, ECDHE P-256, P-384)
5	TLS HMAC Keys	HMAC	TLS integrity and authentication session keys (SHA-1, SHA-256, SHA-384)
6	TLS Encryption Keys	AES	TLS encryption session keys (128 and 256 CBC or GCM)

CSP #	CSP/Key Name	Type	Description
7	SSH Session Authentication Keys	HMAC	Authentication keys used in all SSH connections to the security module's command line interface.(SHA-1)
8	SSH Session Encryption Keys	AES	Used in all SSH connections to the security module's command line interface. (128, 192, and 256 CBC or CTR)
9	SSH DH Private Components	DH	Diffie Hellman private component used in key establishment (DHE 2048)
10	IPsec/IKE authentication Keys	HMAC	(SHA-1, SHA-256, SHA-384 or SHA- 512) Used to authenticate the peer in an IKE/IPsec tunnel connection.
11	IPsec/IKE session Keys	AES	Used to encrypt IKE/IPsec data. These are AES (128, 192, and 256 CBC) IKE keys and (128, 192, and 256 CBC, 128 CCM, 128 and 256 GCM) IPsec keys
12	IPsec/IKE Diffie Hellman Private Components	DH	Diffie-Hellman (Group 14, 19 and 20) private component used in key establishment
13	IPsec pre-shared Keys	Part of HMAC	Manually distributed by an administrator . Used in authentication.
14	IPsec session Keys	AES	(128 CBC, 128 and 256 GCM) Used to encrypt remote access sessions utilizing IPsec.
15	IPsec authentication HMAC	HMAC	(SHA-1) Used in authentication of remote access IPsec data.
16	Firmware code integrity check	HMAC	Used to check the integrity of crypto-related code. (HMAC-SHA-256)
17	Firmware Content Encryption Key	AES-256	Used to decrypt firmware, software, and content.
18	Password	Password	Authentication string with a minimum length of 6 characters.
19	DRBG Seed /State	DRBG	Used by DRBG. The state includes the V and the Key.

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 when not in use. The Master Key is used for encrypting all CSPs in PAN-OS. The Master Key can be configured locally on the firewall or on HSM. AES 256 is utilized for encryption.

The TOE zeroizes 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 or more times using a different alternating data pattern each time.

For volatile memory and non-volatile EEPROM and Flash memories, the zeroization is executed by a single direct overwrite consisting of a pseudo random pattern, followed by a read-verify.

The algorithms used are AES (CBC, GCM) 128, and 256 bit ciphers (AES as specified in ISO 18033-3, CBC as specified in ISO 10116, GCM as specified in ISO 19772), in conjunction with HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384 and HMAC-SHA-512 (see block and digest sizes in **Table 8**), SHA-1, SHA-256, SHA-384 and SHA-512 (digest sizes 160, 256, 384 and 512 bits) and RSA or ECDSA signature verification: see **Table 5**. The implementations are in accordance with FIPS PUB 186-4, “Digital Signature Standard”, ISO/IEC 10118-3:2004 and ISO/IEC 9797-2:2011, Section 7 “MAC Algorithm 2”.

Table 8 HMAC Key Length, Block Size, and Output Length

HMAC	Key Size Range	Key Length (Bits)	Block Size (Bits)	Hash Function	Output MAC Length (Bits)
HMAC-SHA-1	KS<BS	256, 448	512	SHA-1	160
	KS=BS	512	512	SHA-1	160
	KS>BS	1536, 2048	512	SHA-1	160
HMAC-SHA-256	KS<BS	256, 448	512	SHA-256	256
	KS=BS	512	512	SHA-256	256
	KS>BS	1536, 2048	512	SHA-256	256
HMAC-SHA-384	KS<BS	256, 448	1024	SHA-384	384
	KS=BS	1024	1024	SHA-384	384
	KS>BS	1536, 2048	1024	SHA-384	384
HMAC-SHA-512	KS<BS	256, 448	1024	SHA-512	512
	KS=BS	1024	1024	SHA-512	512
	KS>BS	1536, 2048	1024	SHA-512	512

The TOE can be configured as a TLS server for mutual certificate-based authentication for secure connections. To enable certificate-based authentication, the TOE must be configured to use a client certificate profile using the Device > Certificate Management > Certificate Profile tab. The TOE uses SSL/TLS service profiles to specify a certificate and the allowed protocol versions for SSL/TLS services. The TOE uses SSL/TLS for the inbound remote administration traffic on the management (MGT) interface. The key agreement parameters of the server key exchange message consist of the key establishment parameters generated by the TOE: Diffie-Hellman parameters with a key size 2048 bits, ECDSA implementing NIST curves secp256r1, secp384r1, and secp521r1. The TOE denies connections from clients requesting connections using SSL 2.0, SSL 3.0, or TLS 1.0 and shall not establish a trusted

channel if the distinguished name (DN) or Subject Alternative Name (SAN) contained in a certificate does not match the expected identifier for the peer.

The TOE can be configured as a TLS server to permit inbound remote administration traffic (HTTPS) in which the peer initiates handshake and peer authentication is performed via username and password credentials. The TOE can optionally be configured as a TLS server to permit inbound remote administration traffic (HTTPS), in which peer authentication is performed via a certificate. The TOE does not establish the connection if the peer presents an invalid certificate during the handshake. The TOE's HTTPS protocol complies with RFC 2818 and is implemented using TLS 1.2 (RFC 5246) and TLS 1.1 (RFC 4346). The key agreement parameters of the server key exchange message consist of the key establishment parameters generated by the TOE: RSA with key size of 2048 bits and 3072 bits, Diffie-Hellman parameters with a key size 2048 bits, ECDSA implementing NIST curves secp256r1, secp384r1, and secp521r1. The TOE denies connections from clients requesting connections using SSL 2.0, SSL 3.0, or TLS 1.0.

The TOE can be configured as a TLS client for mutual certificate-based authentication for secure communications to the UIA, and the update server. The TOE verifies that the presented identifier matches the reference identifier according to RFC 6125 and only establishes a trusted channel if the peer certificate is valid. The TOE determines certificate validity by verifying the identifier, certificate path, the expiration date, and the revocation status in accordance with RFC 5280. The TOE includes support for client-side certificates for TLS mutual authentication using X.509v3 certificates. The TOE compares the external server's presented identifier to the reference identifier by matching the certificate Common Name (Subject), FQDN (hostname), IP address, . The TOE supports IP address reference identifiers and wildcards for peer authentication. Certificate pinning is not supported. The TOE presents the Supported Elliptic Curves Extension in the Client Hello with the secp256r1, secp384r1, and secp521r1 NIST curves and is enabled by default.

The TOE can be configured as a TLS client for secure communication to an external audit server. The TOE verifies that the presented identifier matches the reference identifier according to RFC 6125 and only establishes a trusted channel if the peer certificate is valid. The TOE compares the external server's presented identifier to the reference identifier by matching the certificate Common Name (Subject), FQDN (hostname), IP address, User FQDN (email address). The TOE supports IP address reference identifiers and wildcards for peer authentication. Certificate pinning is not supported. The TOE presents the Supported Elliptic Curves Extension in the Client Hello with the secp256r1, secp384r1, and secp521r1 NIST curves and is enabled by default.

The TOE implements TLS 1.2 (RFC 5246) and TLS 1.1 (RFC 4346). The TSF supports the following ciphersuites when configured as a TLS Client:

- TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268
- TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492
- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492
- TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246
- TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246
- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289

The TOE implements TLS 1.2 (RFC 5246) and TLS 1.1 (RFC 4346). The TSF supports the following ciphersuites when configured as a TLS Server:

- TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268

- TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268
- TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246
- TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246
- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289

The TOE includes an implementation of IPsec in accordance with RFC 4301. The primary cryptographic algorithms used by the TOE include AES-CBC-128, AES-CBC-256 (both specified by RFC 3602); and AES-GCM-128, AES-GCM-256 as specified in RFC 4106 along with IKEv1 using main mode for Phase 1 exchanges as defined in RFCs 2407, 2408, 2409, RFC 4109, and RFC 4304 for extended sequence numbers; and IKEv2 as defined in RFCs 5996 (with mandatory support for NAT traversal as specified in section 2.23), and 4868 for hash functions. Note that the TOE supports both main and aggressive modes, though aggressive mode should be disabled in the evaluated configuration. The modes can be configured using the GUI to auto, main, or aggressive; the default mode is “auto”. The CC guidance document instructs the administrator to set it “main”. The TOE supports tunnel mode and uses the SHA-based HMAC algorithms as specified in FCS_COP.1(4) Cryptographic Operations (Keyed Hash Algorithm).

The TOE provides mechanisms to implement an IPsec Security Policy Database (SPD) and to process packets to satisfy the behavior of DISCARD, BYPASS and PROTECT packet processing as described in RFC 4301. This is achieved through the administrator configuring appropriately specified access control lists (ACLs). The 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 interzone 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.

Packets matching the destination IP address are permitted otherwise they are denied. The TOE also supports Network Address Translation (NAT) policies 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.

IKEv1 Phase 1 and IKEv2 SA lifetimes are configurable by an authorized administrator and can be specified in seconds, minutes, hours, or days in the range 3 minutes to 8760 hours. IKEv1 Phase 2 and IKEv2 Child SA lifetimes are similarly configurable by an authorized administrator in seconds, minutes, hours, or days in the range 3 minutes to 8760 hours. IKEv1 Phase 2 and IKEv2 Child SA lifetimes can also be established based on number of packets or bytes.

The IKEv1 and IKEv2 protocols implemented by the TOE include DH Group 14 (2048-bit MODP), DH Groups 19 (256-bit Random ECP), and 20 (384-bit Random ECP), using RSA (aka rDSA) and ECDSA peer authentication. In the IKEv1 phase 1 and phase 2 exchanges, the TOE and peer will agree on the best DH group both can support. When the TOE initiates IKE negotiation, the DH group is sent in order according to the peer’s configuration. When the TOE receives an IKE proposal, it will select the first match and the negotiation will fail if there is no match. During IKEv1 phase 1 authentication is based on a verifiable signature as described in RFC2409.

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: 224 (for DH Group 14), 256 (for DH Group 19), 384 (for DH Group 20) bits.

The TOE generates nonces used in IKEv1 and IKEv2 of at least 128 bits in size (half the output size of the negotiated pseudorandom function hash). Nonces are generated using the AES-CTR DRBG implemented by the TOE. The TOE supports PRF hash functions SHA-256, SHA-384, and SHA-512.

The TOE provides AES-CBC-128 and AES-CBC-256 for encrypting IKEv1 and IKEv2 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.

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

- FCS_CKM.1—see table above.
- FCS_CKM.2—see table above.
- FCS_CKM.4—see table above.
- FCS_COP.1(1)—see table above.
- FCS_COP.1(2)—see table above
- FCS_COP.1(3)—see table above.
- FCS_COP.1(4)—see table above.
- FCS_HTTPS_EXT.1—the TOE supports HTTPS web-based secure administrator sessions.
- FCS_IPSEC_EXT.1—The TOE supports IPsec cryptographic network communication protection.
- FCS_RBG_EXT.1—see table above.
- FCS_TLSC_EXT.1- The TOE acts as a TLS client for secure communication with an external audit server. 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.
- FCS_TLSC_EXT.2 – The TOE acts as a TLS client for secure connections with the UIA, WildFire, the Panorama Management System, and the update server. The TOE verifies that the presented identifier matches the reference identifier according to RFC 6125; and only establishes a trusted channel if the peer certificate is valid. The TOE determines certificate validity by verifying the identifier, certificate path, the expiration date, and the revocation status in accordance with RFC 5280. The TOE includes support for client-side certificates for TLS mutual authentication using X509v3 certificates.
- FCS_TLSS_EXT.1 – The TOE acts as a TLS Server when remote administrators connect to the TOE's GUI using HTTPS. The TOE's HTTPS protocol complies with RFC 2818 and is implemented using TLS 1.2 (RFC 5246) and TLS 1.1 (RFC 4346).
- FCS_TLSS_EXT.2 – The TOE and Panorama use SSL/TLS service profiles to specify a certificate and the allowed protocol versions for SSL/TLS services. The TOE and Panorama use SSL/TLS for the inbound traffic on the management (MGT) interface.

6.3 User Data Protection

The TSF allocates and releases the memory resources used for network packet objects. Both when it receives data from the network and when it transmits data to the network, it ensures that the buffers are not padded out with previously transmitted or otherwise residual information by overwriting unused parts of the buffer with 0s.

The User Data Protection security function is designed to satisfy the following security functional requirements:

- FDP_RIP.2—the TOE always overwrites resources when allocated for use in objects.

6.4 Identification and Authentication

The TOE 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 warning banner before authentication.

The TOE maintains user accounts which it uses to control access to the firewall. When creating a new user account, the administrator specifies a user name (i.e., user identity), a password or X509 certificate/common access card, and a role. To enable certificate-based authentication, the TOE must be configured to use a client certificate profile using the Device > Certificate Management > Certificate Profile tab. When a client certificate profile is enabled, each administrator must use a client certificate for access to the TOE via IPsec and TLS. Only one role is specified in the user account per user. The TOE uses the user name and password attributes to identify and authenticate the user when the user logs in via the GUI. With certificate-based authentication, a digital signature is exchanged and verified, in lieu of a password. The TOE does not echo passwords as they are entered. It uses the role attribute to specify user permissions and control what the user can do with the GUI.

The administrator can logon to the GUI by using a secure connection (HTTPS) from a web browser. The administrator enters the IP address of the TOE and their username and password. The TOE also can be configured to require a client certificate (mutual authentication) and additionally require the username and password or not. The credentials may be supplied by a CAC or retrieved from the client computer. The TOE logs all unsuccessful authentication attempts in the System Log.

In order for an administrator to log to the GUI using IPsec, an IPsec tunnel has to be established between the client laptop/management station and the TOE. The administrator uses a third party IPsec client for setting up an IPsec tunnel to the TOE. Authentication is performed using certificates. The administrator runs a web browser and establishes TLS over IPsec. The authentication method for the user can be performed using a password or a certificate. If the user is using a CAC, the card must be inserted into a card reader attached to the user's laptop/management station. Regardless of whether the certificate is on a CAC or not, there is no difference in how the TOE uses the client certificate to authenticate the user besides the use of the CAC reader and accessing the credential on the card.

Regardless of whether a user logs in using an HTTPS or IPsec 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 on the certificate is validated by the TOE.

Passwords can be composed of upper and lower case letters, numbers and special characters. There are no restrictions on any password field character sets. The minimum password length is configurable by the administrator up to a maximum length of 31 characters.

The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for IPsec, and TLS connections. Public key infrastructure (PKI) credentials, such as Rivest, Shamir, and Adelman (RSA) keys and certificates are stored in the TOE's underlying file system on the appliance. Certificates and their associated private key are stored in a single container: the Certificate File. The PKCS#12 file consists of an Encrypted Private Key and X509 Certificate. By default all the private keys are protected since they are always stored in encrypted format using AES-256. The physical security of the appliance (A.PHYSICAL_PROTECTION) protects the appliance and the certificates from being tampered with or deleted. In addition, the TOE identification and authentication security functions protect an unauthorized user from gaining access to the TOE.

The TOE supports Open Certificate Status Protocol (OCSP) and Certificate Revocation List (CRL) status verification for certificate profiles. If both are configured, the devices first try the OCSP method; if the OCSP server is unavailable, the devices use the CRL method.

The TOE uses the following rules for validating the extendedKeyUsage field:

- Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.
- Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.
- OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.

The TOE downloads and caches OCSP status information for every CA listed in the trusted CA list of the firewall. 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 firewall never validated a certificate, the firewall cache does not store the OCSP information for the issuing CA. To use OCSP for verifying the revocation status of certificates, you must configure the firewall to access an OCSP responder (server). The entity that manages the OCSP responder can be a third-party certificate authority (CA) or, if your enterprise has its own public key infrastructure (PKI), the firewall itself.

When establishing an SSL/TLS session, clients can use OCSP to check the revocation status of the authentication certificate. The authenticating client sends a request containing the serial number of the certificate to the OCSP responder (server). The responder searches the database of the certificate authority (CA) that issued the certificate and returns a response containing the status (good, revoked or unknown) to the client. The advantage of the OCSP method is that it can verify status in real-time, instead of depending on the issue frequency (hourly, daily, or weekly) of CRLs.

The TOE downloads and caches the last-issued CRL for every CA listed in the trusted CA list of the firewall. Caching only applies to validated certificates; if a firewall never validated a certificate, the firewall cache does not store the CRL for the issuing CA. Also, the cache only stores a CRL until it expires. The firewall supports CRLs only in Distinguished Encoding Rules (DER) format.

The authorized administrator may generate a self-signed root CA certificate 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 firewall 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 a certificate is part of a chain, the TOE checks the status of every certificate in the chain except the root CA certificate, for which it cannot verify revocation status.

The TOE validates a certificate path by ensuring the presence of the basicConstraints extension is present and the cA flag is set to TRUE for all CA certificates. The TOE forms a Certificate trust path by ensuring that the basic constraints are met, proper key usage parameters exist, the CA flag exists, performing a revocation check of each certificate in the path and performing the validity of the CA certificate. The TOE will not treat a certificate as a CA certificate if the basicConstraints extension is not present or the cA flag is not set to TRUE.

The TOE compares a peer's presented identifier to the reference identifier as follows.

- CAC (Common Access Card) or client certs for authentication of users prior to accessing systems - An x.509 certificate is provided by the user/client upon connecting to a secured resource. Using that certificate, the identity of the user is established and that information is used to determine what level of access should be allowed. If the Subject Alternate name (SAN) is present in the certificate then it is used as a username to perform verification. The TOE performs DNS lookup for usernames that are FQDNs. If the SAN is not present then we use the subject DN in the certificate as the username. This username can then be used to lookup group membership info in a directory located in TOE files. In order to validate the cert, the TOE checks whether the issuing CA is a trusted issuer by PAN-OS. If the client-certificate section is specified and use-crl and/or use-ocsp are specified, the validity of the client certificate will be verified based on the methods specified. The order is always OCSP followed by CRL if both are set. Device authentications occur as follows.
 - For trusted channel connections with remote gateways/peers, the TOE requires the IKE peer id to be configured for certificate authentication: if the type is DN, the TOE checks the peer id against subject DN; otherwise it is checked against the SAN field.
 - Device authentication for the transmission of audit records to an audit server using IPsec or TLS occurs as follows. If the server certificate provided by the audit server has Subject Alternate Name or multiple names (SANs) then each one of those names are verified against the server name/ip configured. If the SAN or SANs is not present in the certificate then the certificate subject DN is checked for a match against the configured server.
 - Connections with the UIA to retrieve the IP address mapping information use TLS 1.2 with RSA_With_AES_256_GCM_SHA384 with hardcoded/predefined, self-signed certificate. The use of pre-defined self-signed internal certs renders the certificate subject name not applicable as it would always be the same.

The TOE will not establish an SA if a certificate or certificate path is deemed invalid; or if the presented identifier does not match the configured reference identifier of the peer as described above. If the TOE cannot establish a connection to determine the validity of a certificate, the administrator may establish the SA or disallow the establishment of the SA.

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

- FIA_UAU_EXT.2.1—the TOE provides local password-based authentication to perform administrative user authentication.
- FIA_UAU.7—the TOE does not echo passwords as they are entered..
- FIA_PMG_EXT.1—the TOE implements a set of password composition constraints as described above.
- FIA_UIA_EXT.1—the TOE displays the warning banner prior to a user being identified and authenticated.
- FIA_X509_EXT.1—the TOE protects, stores and allows authorized administrators to load X.509v3 certificates for use to support authentication.
- FIA_X509_EXT.2—The administrator may establish the SA or disallow the establishment of the SA if the TOE cannot establish a connection to determine the validity of a certificate,
- FIA_X509_EXT.3—the TOE checks the status of every certificate in the chain when a certificate is part of a chain.

6.5 Security Management

The TOE provides a GUI management interface to support security management of the TOE. The GUI is accessible via direct connection to the management port on the device, or remotely over HTTPS or IPsec. The management interfaces enable the authorized administrators to configure the TOE functions and to manipulate TOE data.

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 custom administrator roles: auditadmin, cryptoadmin, and securityadmin. These administrator roles are all considered Security Administrator as defined in the [NDcPP] and [FWcPP] for the purposes of this ST. All roles can administer the TOE both locally and remotely.

The guidance documentation for the evaluated version of the TOE indicates the Superuser role is intended only for initial configuration, to create the administrator accounts for the Security Administrator, Audit Administrator, and Cryptographic Administrator, and that during normal operation the Superuser, Superuser (read-only), Device Administrator, Device Administrator (read-only), Virtual System Administrator, and Virtual System Administrator (read-only) admin roles are not to be assigned to administrators.

- auditadmin—the Audit Administrator is responsible for the regular review of the firewall's audit data.
- cryptoadmin—the Cryptographic Administrator is responsible for the configuration and maintenance of cryptographic elements related to the establishment of secure connections to the firewall.
- securityadmin—the Security Administrator is responsible for all other administrative tasks (e.g. creating the firewall's security policy) not addressed by the other two administrative roles.

The security management functions provided by the TOE are:

- Ability to administer the TOE locally and remotely;
- Ability to configure the access banner;
- Ability to configure the session inactivity time before session termination or locking;
- Ability to update the TOE, and to verify the updates using digital signature capability prior to installing those updates;
- Ability to configure firewall rules;
- Ability to configure the cryptographic functionality;

- Ability to configure the IPsec functionality;
- Ability to import X.509v3 certificates;
- Ability to configure audit behavior;
- Ability to configure the list of TOE-provided services available before an entity is identified and authenticated, as specified in FIA_UIA_EXT.1.

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

- FMT_MOF.1(1)/TrustedUpdate— The initiation of manual updates is restricted to Security Administrators.
- FMT_MOF.1(2)/TrustedUpdate—The Security Administrator to enable /disable the TOE for automatic updates and automatic checking for updates.
- FMT_MOF.1(1)/AdminAct—Security Administrators may modify the behaviour of the functions TOE Security Functions to Security Administrators.
- FMT_MOF.1(2)/AdminAct—The Security Administrators may enable or disable services.
- FMT_MOF.1(1)/Audit –The TOE shall restricts the ability to determine the behaviour of or modify the behaviour of the transmission of audit data to an external IT entity to Security Administrators.
- FMT_MTD.1—the TOE restricts the ability to manage the TSF data to Security Administrators.
- FMT_MTD.1/AdminAct—The TOE restricts the ability to modify, delete, generate/import the cryptographic keys to Security Administrators.
- FMT_SMF.1—the TOE includes the functions necessary to administer the TOE locally and remotely, to manage the cryptomodule and associated functions, and to manage and verify updates of the TOE.
- FMT_SMR.2—the TOE includes three predefined roles that have been configured to access the security management functions of the TOE corresponding to the required ‘Security Administrator’.

6.6 Protection of the TSF

The TOE provides self-tests at start-up (which are also on-demand tests available to administrators) to demonstrate the correct operation of: key error detection, cryptographic algorithms, and RNG. Conditional self-tests are also run during the course of normal operation. The self-tests verify the integrity of stored TSF executable code and TSF data. The TOE performs the following Power-on self-tests:

- AES Encrypt Known Answer Test
- AES Decrypt Known Answer Test
- AES GCM Encrypt Known Answer Test
- AES GCM Decrypt Known Answer Test
- AES CCM Encrypt Known Answer Test
- AES CCM Decrypt Known Answer Test
- RSA Sign Known Answer Test
- RSA Verify Known Answer Test
- ECDSA Sign Known Answer Test
- ECDSA Verify Known Answer Test
- HMAC-SHA-1 Known Answer Test
- HMAC-SHA-256 Known Answer Test
- HMAC-SHA-384 Known Answer Test
- HMAC-SHA-512 Known Answer Test
- SHA-1 Known Answer Test
- SHA-256 Known Answer Test
- SHA-384 Known Answer Test
- SHA-512 Known Answer Test
- DRBG SP800-90A Known Answer Tests

- SP 800-90A Section 11.3 Health Tests
- DH Known Answer Test
- ECDH Known Answer Test
- Firmware Integrity Test – verified with HMAC-SHA-256 and ECDSA P-256. If the calculated result does not equal the previously generated result, the software/firmware test shall fail.

A known-answer test involves operating the cryptographic algorithm on data for which the correct output is already known and comparing the calculated output with the previously generated output (the known answer). If the calculated output does not equal the known answer, the known-answer test shall fail.

The TOE performs the following Conditional Self-Tests within the cryptographic module when the conditions specified for the tests occur:

1. Continuous Random Number Generator (RNG) test – performed on NDRNG and DRBG
2. RSA Pairwise Consistency Test
3. ECDSA Pairwise Consistency Test
4. Firmware Load Test – Verify using RSA 2048 with SHA-256 signature on firmware at time of load. If the digital signature cannot be verified, the test shall fail.

The RNG continuous random number generator test is performed on each RNG and tests for failure to a constant value as follows:

1. If each call to a RNG produces blocks of n bits (where $n > 15$), the first n -bit block generated after power-up, initialization, or reset shall not be used, but shall be saved for comparison with the next n -bit block to be generated. Each subsequent generation of an n -bit block shall be compared with the previously generated block. The test shall fail if any two compared n -bit blocks are equal.
2. If each call to a RNG produces fewer than 16 bits, the first n bits generated after power-up, initialization, or reset (for some $n > 15$) shall not be used, but shall be saved for comparison with the next n generated bits. Each subsequent generation of n bits shall be compared with the previously generated n bits. The test fails if any two compared n -bit sequences are equal.

The TOE performs the following pair-wise consistency tests for public and private keys:

1. If the keys are used to perform an approved key transport method or encryption, then the public key shall encrypt a plaintext value. The resulting ciphertext value shall be compared to the original plaintext value. If the two values are equal, then the test shall fail. If the two values differ, then the private key shall be used to decrypt the ciphertext and the resulting value shall be compared to the original plaintext value. If the two values are not equal, the test shall fail.
2. If the keys are used to perform the calculation and verification of digital signatures, then the consistency of the keys shall be tested by the calculation and verification of a digital signature. If the digital signature cannot be verified, the test shall fail.

Failed self-tests comply with FIPS 140-2 requirements, i.e., a generated key shall not be used, the cryptographic module shall react as required by FIPS PUB 140-2 for failing a self-test, and this event will be audited. If a self-test fails, the TOE enters an error state and outputs an error indicator. The TOE doesn't perform any cryptographic operations while in the error state. All data output from the TOE is inhibited when an error state exists. Should one or more power-up self-tests fail the module will reboot and enter a state in which the reason for the reboot can be determined.

Certificates and their associated private key are stored in a single container: the Certificate File. The PKCS#12 file consists of an Encrypted Private Key and X509 Certificate. By default all the private keys are protected since they are always stored in encrypted format using AES-256. The TOE prevents the reading of all keys by encrypting them with a Master Key using AES-256. The TOE does not provide an interface to read the Master Key. The TOE is designed specifically to prevent access to locally-stored cryptographically protected passwords and does not disclose any keys stored in the TOE. The TOE protects the confidentiality of user passwords by encrypting the password using AES-256. The TOE does not offer any functions that will disclose to any users a stored cryptographic key or password.

The TOE is a hardware appliance or a virtual appliance image installed on a virtualization platform that includes a hardware-based real-time clock. The hardware hosting the PAN-OS VMs provides the time clock, as well as CPU,

ports, etc., which are provided by VM environment (hypervisor). The TOE's embedded OS manages the clock and exposes administrator clock-related functions. The clock is used for audit record time stamps, measuring session activity for termination, and for cryptographic operations based on time/date. Authorized administrators may query the current software/firmware version of the TOE. Note that the TOE is firmware and software. When updates are available from Palo Alto, an administrator can obtain and install those updates from <https://updates.paloaltonetworks.com>. The secured connection to the Palo Alto server supports TLS v1.1, TLS v1.2 and uses FIPS-approved algorithms. For an additional layer of protection, Palo Alto Networks has chosen to sign (using RSA-2048) and encrypt (using AES-256) all content that is downloaded to the firewall. The TOE update package and its corresponding digital signature are downloaded from the Palo Alto support site directly onto the appliance, or downloaded to another computer and then upload it to the appliance. The integrity check is performed by verifying the signature using the public key (corresponding to the RSA key used to create the signature) as part of the process of loading the image onto the TOE. This makes the image available for installation, but an install (activation) is not initiated automatically. The administrator can view which versions of the TOE software have been downloaded and which is the currently running version and can choose to install/activate an update from this screen. Certificates and keys are stored on the TOE's file system.

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

- FPT_APW_EXT.1—the TOE does not offer any functions that will disclose to any user a plain text password. Note that passwords are stored encrypted with a Master Key using AES-256.
- FPT_SKP_EXT.1—the TOE does not offer any functions that will disclose to any users a stored cryptographic key.
- FPT_STM.1—the TOE provides its own reliable time stamps for its own use.
- FPT_TST_EXT.1—the TOE includes self-tests at start-up (which are also on-demand tests available to administrators) on all cryptographic functions. Conditional self-tests are also run during the course of normal operation.
- FPT_TST_EXT.2—The firmware integrity is verified using HMAC-SHA-256 and ECDSA P-256.
- FPT_TUD_EXT.1—the administrator may query the currently executing version of the TOE software and initiate software/firmware updates for the TOE. The download is verified using a digital signature.

6.7 TOE Access

The TOE can be configured to display an informative banner that will appear prior to authentication when accessing the TOE via either a direct or remote connection to the management port in order to access the Web Interface (GUI). The TOE subsequently will enforce an administrator-defined inactivity timeout value after which the inactive session will be terminated.

The TOE can be configured by an administrator to set an interactive session timeout value (any integer value from 1 to 60 minutes) and also optionally in seconds. The function is disabled by default and the administrator must follow the guidance to configure the session timeout value. A remote session that is inactive (i.e., no commands issuing from the remote client) for the defined timeout value will be terminated. A local session that is similarly inactive for the defined timeout period will be terminated. The user will be required to re-enter their user id and their password so they can establish a new session once a session is terminated. If the user id and password match those of the user that was locked, the session is reconnected with the console and normal input/output can again occur for that user.

The TOE provides both local and remote users the ability to logout (or terminate) their sessions as directed by the user.

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

- FTA_SSL.3—the TOE terminates remote sessions that have been inactive for an administrator-configured period of time.
- FTA_SSL.4—the TOE provides the function to logout (or terminate) both local and remote user sessions as directed by the user.

- FTA_SSL_EXT.1—the TOE terminates local sessions that have been inactive for an administrator-configured period of time.
- FTA_TAB.1—the TOE can be configured to display administrator-defined advisory banners before establishing an administrative user session.

6.8 Trusted path/channels

The TOE can be configured to export audit records to an external Syslog server using IPsec or TLS. The TOE uses TLS to protect communications between itself and the UIA, connections to Wildfire, Panorama and with the update server for TOE updates. The TOE can be instructed to contact Palo Alto Networks' update server to download new content or TOE software updates.

To support secure remote administration, the TOE includes an implementation of HTTPS and supports IPsec. An authorized administrator can establish secure remote connections with the TOE using HTTP over TLS or by establishing an IPsec connection. To successfully establish an interactive administrative session, the administrator must be able to provide acceptable user credentials (e.g., certificate; or user id, password, and role), after which they will be able to access the GUI features. The TOE requires the use of the trusted path for initial administrator authentication and all subsequent remote administrative actions.

The secure protocols are supported by NIST-validated cryptographic mechanisms included in the TOE implementation.

The Trusted Path/Channels security function is designed to satisfy the following security functional requirements:

- FTP_ITC.1—the TOE can be configured to ensure that exported audit records are sent only to the configured Syslog server using IPsec or TLS so they are not subject to inappropriate disclosure or modification. The TOE uses TLS for the communication channel between itself and the UIA, connections to Wildfire, Panorama and with the update server for TOE updates. The TOE permits the TSF to initiate communication with the Syslog server and the update server, and the authorized IT entities to initiate communication using either TLS or the IPsec trusted channel.
- FTP_TRP.1—the TOE provides IPsec and HTTP over TLS to support secure remote administration. Administrators can initiate a remote session that is secured (from disclosure and modification) using NIST-validated cryptographic operations, and all remote security management functions require the use of this secure channel.

6.9 Stateful Traffic Filtering

An authorized administrator may configure the TOE to apply stateful traffic filtering rules of permit, deny, and log on the following protocols:

- Internet Control Message Protocol version 4 (ICMPv4)
- Internet Control Message Protocol version 6 (ICMPv6)
- Internet Protocol (IPv4)
- Internet Protocol version 6 (IPv6)
- 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.

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 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, applying the first rule that matches the incoming traffic.

Security policies can be defined only between zones of the same type. 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:
 - 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 deny 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 ‘deny’
- 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 deny 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 administrator may limit the number of half-open TCP connections and defines the thresholds that constitute flooding. In general, 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.

A DoS Protection policy rule that is set to protect (rather than to allow or deny 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 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 or UDP 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. Logging can be enabled in the security policy rule configured to control the FTP traffic.

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 IPv4 packets with the shared address space address range

100.64.0.0/10 as specified in RFC 6598. 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.
- drop all inbound and outbound IPv6 packets containing a Hop-by-Hop header with option type values intended for Destination Options.
- 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)
 - The source user²¹ (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 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).

²¹ Source user in policies is not within the scope of the evaluation.

- 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 Network 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. The timeout period due to inactivity is administrator configurable from 1 – 6044800 seconds. 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 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 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 dataplane) 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.

The Stateful Traffic Filtering security function is designed to satisfy the following security functional requirements:

- FFW_RUL_EXT.1—an authorized administrator may configure the TOE to apply stateful traffic filtering rules of permit, deny, and log on the following protocols: ICMPv4, ICMPv6, IPv4, IPv6, TCP, UDP.
- FFW_RUL_EXT.2: Stateful Filtering of Dynamic Protocols

7. Protection Profile Claims

This ST is conformant to the *collaborative Protection Profile for Stateful Traffic Filter Firewalls, Version 1.0, 27 February 2015* [FWcPP], and the *collaborative Protection Profile for Network Devices, Version 1.0, 27 February 2015* [NDcPP].

The TOE is a stateful traffic filter firewall appliance. As such, the TOE is a network device making the [NDcPP], and [FWcPP] claims valid and applicable.

As explained in section 3, Security Problem Definition, the Security Problem Definitions of the [NDcPP] and [FWcPP] and have been included by reference into this ST.

As explained in section 4, Security Objectives, the Security Objectives of the [NDcPP] and [FWcPP] and have been included by reference into this ST.

The following table identifies all the Security Functional Requirements (SFRs) in this ST. Each SFR is reproduced from the [NDcPP] and [FWcPP], and operations completed as appropriate. The source is determined first by any applicable TDs and second by PP.

Table 9 SFR Protection Profile Sources

Requirement Class	Requirement Component	Source
FAU: Security audit	FAU_GEN.1: Audit Data Generation	NDcPP FWcPP
	FAU_GEN.2: User identity association	NDcPP FWcPP
	FAU_STG.1: Protected audit trail storage	FWcPP
	FAU_STG_EXT.1: Protected Audit Event Storage	FWcPP
	FAU_STG_EXT.3: Display warning for local storage space	FWcPP
FCS: Cryptographic support	FCS_CKM.1: Cryptographic Key Generation	NDcPP FWcPP
	FCS_CKM.2: Cryptographic Key Establishment	NDcPP FWcPP
	FCS_CKM.4: Cryptographic Key Destruction	NDcPP FWcPP
	FCS_COP.1(1): Cryptographic Operation (AES Data Encryption/Decryption)	NDcPP FWcPP
	FCS_COP.1(2): Cryptographic Operation (Signature Generation and Verification)	NDcPP FWcPP
	FCS_COP.1(3): Cryptographic Operation (Hash Algorithm)	NDcPP FWcPP
	FCS_COP.1(4): Cryptographic Operation (Keyed Hash Algorithm)	NDcPP FWcPP
	FCS_HTTPS_EXT.1: HTTPS Protocol	NDcPP FWcPP
	FCS_IPSEC_EXT.1: IPsec Protocol	NDcPP FWcPP
	FCS_RBG_EXT.1: Random Bit Generation	NDcPP FWcPP
	FCS_TLSC_EXT.1 - TLS Client Protocol	NDcPP FWcPP
	FCS_TLSC_EXT.2 - TLS Client Protocol with authentication	NDcPP FWcPP
	FCS_TLSS_EXT.1 - TLS Server Protocol	NDcPP FWcPP
	FCS_TLSS_EXT.2 - TLS Server Protocol with mutual authentication	NDcPP FWcPP

Requirement Class	Requirement Component	Source
FDP: User data protection	FDP_RIP.2: Full Residual Information Protection	NDcPP FWcPP
FIA: Identification and authentication	FIA_PMG_EXT.1: Password Management	NDcPP FWcPP
	FIA_UAU.7: Protected Authentication Feedback	NDcPP FWcPP
	FIA_UAU_EXT.2: Extended: Password-based Authentication Mechanism	NDcPP FWcPP
	FIA_UIA_EXT.1: User Identification and Authentication	NDcPP FWcPP
	FIA_X509_EXT.1: X.509 Certificate Validation	NDcPP FWcPP
	FIA_X509_EXT.2: X.509 Certificate Authentication	NDcPP FWcPP
	FIA_X509_EXT.3: X.509 Certificate Requests	NDcPP FWcPP
FFW: Stateful Traffic Filtering	FFW_RUL_EXT.1: Stateful Traffic Filtering	FWcPP
	FFW_RUL_EXT.2: Stateful Filtering of Dynamic Protocols	FWcPP
FMT: Security Management	FMT_MOF.1(1)/TrustedUpdate: Management of security functions behaviour	NDcPP FWcPP
	FMT_MOF.1(2)/TrustedUpdate: Management of Security Functions Behavior	NDcPP FWcPP
	FMT_MOF.1(1)/AdminAct: Management of security functions behaviour	NDcPP FWcPP
	FMT_MOF.1(2)/AdminAct: Management of Security Functions Behavior	NDcPP FWcPP
	FMT_MOF.1(1)/Audit: Management of security functions behavior	NDcPP FWcPP
	FMT_MTD.1: Management of TSF Data	NDcPP FWcPP
	FMT_MTD.1/AdminAct: Management of TSF Data	NDcPP FWcPP
	FMT_SMF.1: Specification of Management Functions	NDcPP FWcPP
	FMT_SMR.2: Restrictions on Security Roles	NDcPP FWcPP
FPT: Protection of the TSF	FPT_SKP_EXT.1: Protection of TSF Data (for reading of all symmetric keys)	NDcPP FWcPP
	FPT_APW_EXT.1: Protection of Administrator Passwords	NDcPP FWcPP
	FPT_STM.1: Reliable Time Stamps	NDcPP FWcPP
	FPT_TST_EXT.1: TSF Testing	NDcPP FWcPP
	FPT_TST_EXT.2: Extended: TSF Testing	NDcPP FWcPP
	FPT_TUD_EXT.1: Extended: Trusted Update	NDcPP FWcPP
FTA: TOE access	FTA_SSL.3: TSF-initiated Termination	NDPP FWcPP
	FTA_SSL.4: User-initiated Termination	NDcPP FWcPP
	FTA_SSL_EXT.1: TSF-initiated Session Locking	NDcPP

Requirement Class	Requirement Component	Source
		FWcPP
	FTA_TAB.1: Default TOE Access Banners	NDcPP FWcPP
FTP: Trusted path/channels	FTP_ITC.1: Inter-TSF trusted channel	NDcPP FWcPP
	FTP_TRP.1: Trusted Path	NDcPP FWcPP

8. Rationale

This security target includes by reference the [NDcPP] and [FWcPP] Security Problem Definition, Security Objectives, and Security Assurance Requirements. The security target makes no additions to the [NDcPP] and [FWcPP] assumptions. Security functional requirements have been reproduced with the protection profile operations completed. Operations on the security requirements follow [NDcPP] and [FWcPP] application notes and assurance activities. Consequently, [NDcPP] and [FWcPP] rationale applies but is incomplete. The TOE Summary Specification rationale below serves to complete the rationale required for the security target.

8.1 TOE Summary Specification Rationale

Each subsection in Section 6, the TOE Summary Specification, describes a security function of the TOE. Each description is followed with rationale that indicates which requirements are satisfied by aspects of the corresponding security function. The set of security functions work together to satisfy all of the security functions and assurance requirements. Furthermore, all of the security functions are necessary in order for the TSF to provide the required security functionality.

This Section in conjunction with Section 6, the TOE Summary Specification, provides evidence that the security functions are suitable to meet the TOE security requirements. The collection of security functions work together to provide all of the security requirements. The security functions described in the TOE summary specification are all necessary for the required security functionality in the TSF. **Table 10 Security Functions vs. Requirements Mapping** demonstrates the relationship between security requirements and security functions.

Table 10 Security Functions vs. Requirements Mapping

	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
FAU_GEN.1	X									
FAU_GEN.2	X									
FAU_STG.1	X									
FAU_STG_EXT.1	X									
FCS_CKM.1		X								
FCS_CKM.2		X								
FCS_CKM.4		X								
FCS_COP.1(1)		X								
FCS_COP.1(2)		X								
FCS_COP.1(3)		X								
FCS_COP.1(4)		X								
FCS_HTTPS_EXT.1		X								
FCS_IPSEC_EXT.1		X								
FCS_RBG_EXT.1		X								
FCS_TLSC_EXT.1		X								
FCS_TLSC_EXT.2		X								
FCS_TLSS_EXT.1		X								
FCS_TLSS_EXT.2		X								
FDP_RIP.2			X							
FIA_PMG_EXT.1				X						
FIA_UAU.7				X						
FIA_UAU_EXT.2				X						
FIA_UIA_EXT.1				X						
FIA_X509_EXT.1				X						
FIA_X509_EXT.2				X						
FIA_X509_EXT.3				X						
FMT_MOF.1(1)/TrustedUpdate:					X					
FMT_MOF.1(1)/Audit:					X					
FMT_MTD.1					X					

	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
FMT_MTD.1/AdminAct:					X					
FMT_SMF.1					X					
FMT_SMR.2					X					
FPT_APW_EXT.1						X				
FPT_SKP_EXT.1						X				
FPT_STM.1						X				
FPT_TST_EXT.1						X				
FPT_TUD_EXT.1						X				
FTA_SSL.3							X			
FTA_SSL.4							X			
FTA_SSL_EXT.1							X			
FTA_TAB.1							X			
FTP_ITC.1								X		
FTP_TRP.1								X		
FFW_RUL_EXT.1									X	
FFW_RUL_EXT.2									X	