Cisco FTD 6.4 on Firepower 1000 and 2100 Series with FMC/FMCv

Security Target

ST Version 1.0

June 10, 2021

Table of Contents

List of Tables

List of Figures

List of Acronyms

The following acronyms and abbreviations are common and may be used in this Security Target:

Table 1: Acronyms

Acronyms/Abbreviations	Definition
ACL	Access Control List
AES	Advanced Encryption Standard
CC	Common Criteria
CEM	Common Evaluation Methodology
CM	Configuration Management
DHCP	Dynamic Host Configuration Protocol
EAL	Evaluation Assurance Level
EHWIC	Ethernet High-Speed WAN Interface Card
ESP	Encapsulating Security Payload
FTD	Firepower Threat Defense
Gbps	Gigabits per second
GE	Gigabit Ethernet port
HTTPS	Hyper-Text Transport Protocol Secure
ICMP	Internet Control Message Protocol
IKE	Internet Key Exchange
IPS	Intrusion Prevention System
IPsec	Internet Protocol Security
IT	Information Technology
NDcPP	Network Device Collaborative Protection Profile
NGIPS	Cisco Next-Generation IPS
OS	Operating System
PoE	Power over Ethernet
POP3	Post Office Protocol
PP	Protection Profile
SA	Security Association
SFP	Small–form-factor pluggable port
SHA	Secure Hash Algorithm
SIP	Session Initiation Protocol
SSHv2	Secure Shell (version 2)
SSM	Security Services Module
SSP	Security Services Processor
ST	Security Target
TCP	Transport Control Protocol
TOE	Target of Evaluation
TSC	TSF Scope of Control
TSF	TOE Security Function
TSP	TOE Security Policy
UDP	User Datagram Protocol
VLAN	Virtual Local Area Network
VPN	Virtual Private Network
WAN	Wide Area Network

Acronyms/Abbreviations	Definition
WIC	WAN Interface Card

DOCUMENT INTRODUCTION

Prepared By:

Cisco Systems, Inc.
170 West Tasman Dr.
San Jose, CA 95134

This document provides the basis for an evaluation of a specific Target of Evaluation (TOE), the Firepower Threat Defense (FTD) with Firepower Management Center (FMC). This Security Target (ST) defines a set of assumptions about the aspects of the environment, a list of threats that the product intends to counter, a set of security objectives, a set of security requirements, and the IT security functions provided by the TOE which meet the set of requirements. Administrators of the TOE will be referred to as administrators, authorized administrators, TOE administrators, semi-privileged, privileged administrators, and security administrators in this document.

1 SECURITY TARGET INTRODUCTION

The Security Target contains the following sections:

- ♦ Security Target Introduction [Section 1]
- ♦ Conformance Claims [Section 2]
- ♦ Security Problem Definition [Section 3]
- ♦ Security Objectives [Section 4]
- ♦ IT Security Requirements [Section 5]
- ♦ TOE Summary Specification [Section 6]
- ♦ Supplemental TOE Summary Specification Information [Section 7]
- ♦ References [Section 8]

The structure and content of this ST comply with the requirements specified in the Common Criteria (CC), Part 1, Annex A, and Part 2.

1.1 ST and TOE Reference

This section provides information needed to identify and control this ST and its TOE.

Table 2: ST and TOE Identification

Name	Description
ST Title	Cisco FTD 6.4 on Firepower 1000 and 2100 Series with FMC/ FMCv Security Target
ST Version	1.0
Publication Date	June 10, 2021
Vendor and ST Author	Cisco Systems, Inc.
TOE Reference	Cisco FTD 6.4 on Firepower 1000 and 2100 Series with FMC/FMCv
TOE Hardware Models	• Firepower 1000 Series (1010, 1120, 1140) • Firepower 2100 Series (2110, 2120, 2130, 2140) • Cisco Firepower Management Center (FMC) (FMC1000-K9, FMC2500-K9, FMC4500-K9, FMC1600-K9, FMC2600-K9 and FMC4600-K9) • FMCv running on ESXi 6.0 or 6.5 on the Unified Computing System (UCS) UCSB-B200-M4, UCSC-C220-M4S, UCSC-C240-M4SX, UCSC-C240-M4L, UCSB-B200-M5, UCSC-C220-M5, UCSC-C240-M5, UCS-E160S-M3 and UCS-E180D-M3
TOE Software Version	FTD 6.4 and FMC/FMCv 6.4
Keywords	Router, IPS/IDS

1.2 TOE Overview

The Cisco Firepower 2100 Series and 1000 Series Appliances are purpose-built, firewall platforms with Intrusion Prevention capabilities provided by Firepower Threat Defense (FTD) software. The Firepower Management Center (FMC) physical and FMCv virtual appliances provide a centralized management console and event database for the FTD, and aggregate and correlate intrusion, discovery, and connection data from the FTD. In this deployment, the FTD provides VPN, network analysis, intrusion detection, and access control functionalities. The TOE includes one or more FTD appliances that are centrally managed by a Firepower Management Center (FMC) appliance, and together the FMC and FTD appliances form the TOE (Distributed TOE Use Case 3). The TOE includes the hardware models as defined in Table 2 of section 1.1.

1.2.1 TOE Product Type

The TOE consists of hardware and software that provide connectivity and security services on multiple secure devices.

The TOE provides intrusion prevention system (IPS) capabilities by combining the security of a Next Generation IPS (NGIPS) with the power of access control, malware protection, and URL/IP filtering (white/black listing) known as Security Intelligence. The TOE monitors incoming and outgoing network traffic and performs real-time traffic analysis and logging using the industry-leading Snort® engine. All packets on the monitored network are scanned, decoded, preprocessed and compared against a set of rules to determine whether inappropriate traffic, such as system attacks, is being sent over the network. The system generates alerts or blocks the traffic when deviations of the expected network behavior are detected or when there is a match to a known attack pattern.

1.2.2 Supported non-TOE Hardware/ Software/ Firmware

The TOE supports the following hardware, software, and firmware in its environment when the TOE is configured in its evaluated configuration:

Table 3: IT Environment Components

Component	Required	Usage/Purpose Description for TOE performance
Management Workstation with SSH Client	Yes	This includes any IT Environment Management workstation with SSH client installed that is used by the TOE administrator to support TOE administration through SSHv2 protected channels. Any SSH client that supports SSHv2 may be used.
Management Workstation with Web Browser	Yes	This includes any IT Environment Management workstation with a web browser installed that is used by the TOE administrator to support TOE administration through TLS/HTTPS protected channels. Any browser that supports TLSv1.1 and TLSv1.2 may be used.
Audit (syslog) Server	Yes	This includes any syslog server to which the TOE would transmit syslog messages. Connections to remote audit servers must be tunneled in IPsec or TLS.

Component	Required	Usage/Purpose Description for TOE performance
Certification Authority	Yes	This includes any IT Environment Certification Authority on the TOE network. This can be used to provide the TOE with a valid certificate during certificate enrollment.

1.3 TOE DESCRIPTION

This section provides an overview and description of the TOE. The TOE is comprised of both software and hardware. The TOE appliances are comprised of the following: Firepower 1000 Series (1010, 1120, 1140), Firepower 2100 Series (2110, 2120, 2130, 2140) and Firepower Management Center (FMC) (FMC1000-K9, FMC2500-K9, FMC4500-K9, FMC1600-K9, FMC2600-K9, FMC4600-K9 and FMCv). The software is comprised of the FTD software image Release 6.4 and FMC (or FMCv) version 6.4.

The models that comprise the TOE have common hardware characteristics. These differing characteristics affect only non-TSF relevant functionality (such as throughput, processing speed, number and type of network connections supported, number of concurrent connections supported, and amount of storage) and therefore support security equivalency of the FTDs in terms of hardware.

Figure 1: FP2100 Series Hardware (2110, 2120, 2130, 2140)

The FP2100 models have the following distinct characteristics:

Table 4: FP2100 Series Hardware

Model	FP2110	FP2120	FP2130	FP2140
Number of Processors	1	1	1	1
Processor(s)	Intel Xeon D-1526 (Broadwell)	Intel Xeon D-1528 (Broadwell)	Intel Xeon D-1548 (Broadwell)	Intel Xeon D-1577 (Broadwell)
	OCTEON III CN7230 MIPS64	OCTEON III CN7340 MIPS64	OCTEON III CN7350 MIPS64	OCTEON III CN7360 MIPS64
Storage	100 GB	100 GB	200 GB	200 GB
Integrated I/O	12 x 10M/100M/1GBASE-T Ethernet interfaces (RJ-45), 4 x 1 Gigabit	12 x 10M/100M/1GBASE-T Ethernet interfaces (RJ-45), 4 x 1 Gigabit	12 x 10M/100M/1GBASE-T Ethernet interfaces (RJ-45), 4 x 10 Gigabit (SFP+) Ethernet interfaces	

	(SFP) Ethernet interfaces	(SFP) Ethernet interfaces		
IPsec VPN Throughput	750 Mbps	1000 Mbps	1500 Mbps	3000 Mbps
Maximum concurrent sessions	1 million	1.2 million	2 million	3.0 million
Maximum VPN Peers	1500	3500	7500	10000

Figure 2: FP1000 Series Hardware (1010)

Figure 3: FP1000 Series Hardware (1120, 1140)

The FP1000 models have the following distinct characteristics:

Table 5: FP1000 Series Hardware

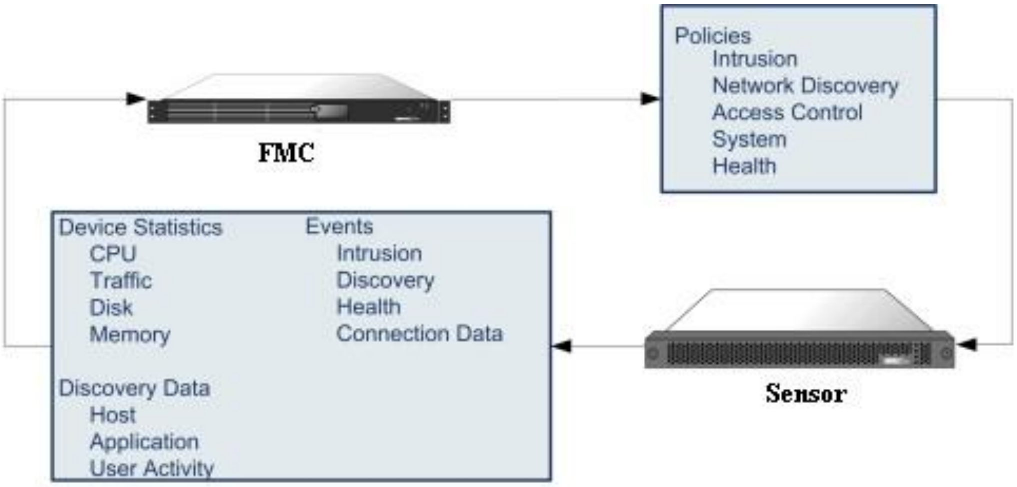
Model	**FP1010**	**FP1120**	**FP1140**
Number of Processors	1	1	1
Processor(s)	Intel Atom C3558 (Goldmont)	Intel Atom C3858 (Goldmont)	Intel Atom C3958 (Goldmont)
Storage	1 x 200 GB	1 x 200 GB	1 x 200 GB
Integrated I/O	8 x RJ45, 4x SFP	8 x RJ45, 4x SFP	8 x RJ45, 4x SFP
IPsec VPN Throughput	300 Mbps	1 Gbps	1.2 Gbps

Maximum concurrent sessions	100K	200K	400K
Maximum VPN Peers	75	150	400

The FMC is a fault-tolerant, purpose-built network appliance that provides a centralized management console and database repository for the Sensors (i.e., FTD). The FMC is a key component in the Cisco NGIPS system. Administrators can use the FMC to manage the full range of Sensors that comprise the NGIPS system, and to aggregate, analyze, and respond to the threats they detect on their network. By using the FMC to manage Sensors, administrators can:

- Configure policies for all Sensors from a single location, making it easier to change configurations.
- Install various types of software updates on Sensors.
- Push policies to managed Sensors and monitor their health status from the FMC.

The FMC aggregates and correlates intrusion events, anomaly, network discovery information, and Sensor performance data, allowing administrators to monitor the information the Sensors are reporting in relation to one another, and to assess the overall activity occurring on their network. The following illustration lists what is transmitted between FMC and its managed Sensors.


FMC
Policies
Intrusion
Network Discovery
Access Control
System
Health
Sensor
Device Statistics
CPU
Traffic
Disk
Memory
Events
Intrusion
Discovery
Health
Connection Data
Discovery Data
Host
Application
User Activity

The FMC hardware components in the TOE have the following distinct characteristics:

Table 6: FMC Models

Model	FMC1000-K9	FMC1600-K9	FMC2500-K9	FMC2600-K9	FMC4500-K9	FMC4600-K9
Processor	Intel Xeon E5-2640 v4 (Broadwell)	Intel Xeon Silver 4110 (Skylake)	Intel Xeon E5-2640 v4 (Broadwell)	Intel Xeon Silver 4110 (Skylake)	Intel Xeon E5-2620 v4 (Broadwell)	Intel Xeon Silver 4116 (Skylake)
Memory	32 GB	32 GB	64 GB	64 GB	128 GB	128 GB
Maximum Number of Sensors Managed	50	50	300	300	750	750
Maximum Number of IPS Events	60 Million	60 Million	60 Million	60 Million	300 Million	300 Million
Event Storage	900 GB	900 GB	1.8 TB	1.8 TB	3.2 TB	3.2 TB
Maximum Flow Rate	6,000 fps	6,000 fps	10,000 fps	10,000 fps	20,000 fps	20,000 fps
Maximum Network Map (hosts/users)	50,000/50,000	50,000/50,000	300,000/300,000	300,000/300,000	600,000/600,000	600,000/600,000
Network Interfaces	2 x 1Gbps	2 x 1Gbps	2 x 1Gbps	2 x 1Gbps	2 x 1Gbps 2 x 10Gbps	2 x 1Gbps 2 x 10Gbps

The underlying Cisco UCS hardware platforms within the TOE have common hardware characteristics. These differing characteristics affect only non-TSF relevant functionality (such as throughput, processing speed, number and type of network connections supported, number of concurrent connections supported, and amount of storage) and therefore support security equivalency of the FMCv in terms of hardware.

Figure 4: UCS Hardware

The UCS hardware components in the TOE have the following distinct characteristics:

Table 7: UCS Hardware

Model	B200 M4	B200 M5	C220 M4S	C220 M5	C240 M4SX	C240 M4L	C240 M5
Number of Processors	2	2	2	2	2	2	2
Processor	Intel® Xeon® E5-2620 v3 (Haswell), Intel Xeon E5-2609v4 (Broadwell)	Intel® Xeon® Bronze 3104 (Skylake), Intel® Xeon® Silver 4110 (Skylake)	Intel® Xeon® E5-2620 v3 (Haswell), Intel Xeon E5-2609v4 (Broadwell)	Intel® Xeon® Bronze 3104 (Skylake), Intel® Xeon® Silver 4110 (Skylake)	Intel® Xeon® E5-2620 v3 (Haswell), Intel Xeon E5-2609v4 (Broadwell)	Intel® Xeon® E5-2620 v3 (Haswell), Intel Xeon E5-2609v4 (Broadwell)	Intel® Xeon® Bronze 3104 (Skylake), Intel® Xeon® Silver 4110 (Skylake)

		Intel® Xeon® Gold 6128 (Skylake) Intel® Xeon® Platinum 8153 (Skylake)		Intel® Xeon® Gold 6128 (Skylake) Intel® Xeon® Platinum 8153 (Skylake)			Intel® Xeon® Gold 6128 (Skylake) Intel® Xeon® Platinum 8153 (Skylake)
Form factor	Half-width blade	Half-width blade	Half-width blade	1RU rack server	Half-width blade	Half-width blade	2 RU
Maximum Memory	1.5 TB, 24 DIMMs	3.0 TB, 24 x DDR4 DIMMs	3.0 TB, 24 x DDR4 DIMMs	3 TB, 24 x DDR4 DIMMs	3.0 TB, 24 x DDR4 DIMMs	3.0 TB, 24 x DDR4 DIMMs	3 TB, 24 x DDR4 DIMMs
Disk Space	3.2 TB	20.5 TB	20.5 TB	80 TB	20.5 TB	20.5 TB	197.6 TB
Max I/O per blade	80 Gbps (2 x 40 Gbps)	80 Gbps (2 x 40 Gbps)	80 Gbps (2 x 40 Gbps)	Undisclosed	80 Gbps (2 x 40 Gbps)	80 Gbps (2 x 40 Gbps)	Undisclosed

Model	**E160S M3**	**E180D M2/K9**
Number of Processors	1	1
Processor	Intel® Xeon® D-1528 (Broadwell)	Intel® Xeon® D-1548 (Broadwell)
Form factor	1 RU	2 RU
Memory	8 – 64 GB	16 – 128 GB
Disk Space	4 TB	4 TB
I/O	● 2 internal Gigabit Ethernet ports (Broadcom 5719) ● 2 external 10 Gigabit Ethernet ports (1000/10000) (Integrated within Intel Xeon-D CPU)	● 2 internal Gigabit Ethernet ports (Broadcom 5719) ● 2 external 10 Gigabit Ethernet ports (1000/10000) (Integrated within Intel CPU)

		● 1 dedicated management Ethernet port (10/100/1000) for Cisco IMC

1.4 TOE Evaluated Configuration

The TOE consists of one or more physical or virtual devices specified in Section 1.5 below and includes the FTD and FMC software. Each instantiation of the TOE has two or more network interfaces and is able to filter IP traffic to and through those interfaces.

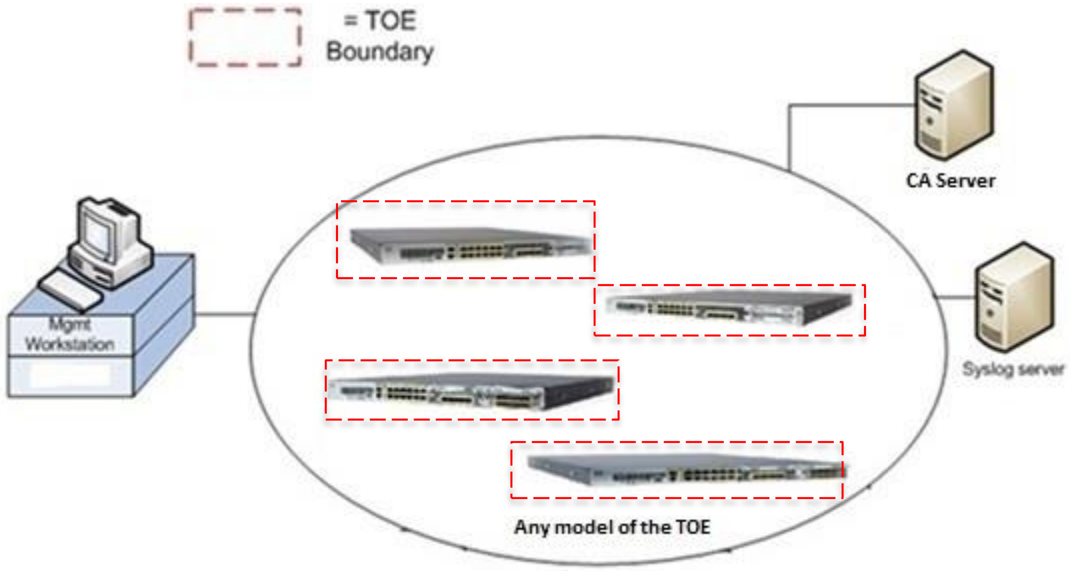
If the TOE is to be remotely administered, the management station must connect using SSHv2. When the Web UI is used, a remote workstation with a TLS-enabled browser must be available. A syslog server can also be used to store audit records, and the syslog server must support syslog over TLS and IPsec.

FTD supports two different TLS clients that send syslog messages to the external syslog server- FTD TLS client and FTD OS TLS Client. The FTD TLS Client is configured by the FMC and is the main audit system for audits generated by FTD. It sends audit events such as IPsec and login messages to the external syslog server. The FTD OS TLS client implementation is configured through the FTD's command line and sends audit events such as SSH login, console login, etc. to an external syslog server.

The TOE is able to filter connections to/from these external entities using its IP traffic filtering, and can encrypt traffic where necessary using TLS, SSH, and/or IPsec. The communication between the FMC and FTD is protected by TLSv1.2. The TOE uses X.509v3 certificates to support authentication for both IPsec and TLS, and the CA server in the Operational environment can be used to obtain digital certificates.

The following figure provides a visual depiction of an example TOE deployment. The TOE boundary is surrounded with a hashed red line.

Figure 5: Example TOE Deployment



The figure includes the following:

- TOE components (at least one FTD and FMC)
- Management Workstation (Operational Environment)
- CA Server (Operational Environment)
- Syslog server (Operational Environment)

1.5 Physical Scope of the TOE

The TOE is a hardware and software solution comprised of the components described in Table 8:

Table 8: Hardware Models and Specifications

TOE Configuration	Hardware Configurations	Software Version
FP2110 **FP2120** **FP2130** **FP2140**	The Cisco FP2100 provides high-performance firewall and VPN services and 4-12 Gigabit Ethernet interfaces, and support for up to 10,000 VPNs.	FTD v6.4
FP1010 **FP1120** **FP1140**	The Cisco FP1000 provides high-performance firewall and VPN services and support for up to 400 VPNs.	FTD v6.4

FMC1000-K9 **FMC2500-K9** **FMC4500-K9** **FMC1600-K9** **FMC2600-K9** **FMC4600-K9**	See table 6 above	FMC v6.4
FMCv	FMCv running on ESXi 6.0 or 6.5 on the Unified Computing System (UCS) UCSB-B200-M4, UCSC-C220-M4S, UCSC-C240-M4SX, UCSC-C240-M4L, UCSB-B200-M5, UCSC-C220-M5, UCSC-C240-M5, UCS-E160S-M3 and UCS-E180D-M3	FMCv v6.4

1.6 Logical Scope of the TOE

The TOE is comprised of several security features including Intrusion Prevention. Each of the security features identified above consists of several security functionalities, as identified below.

1. Security Audit
2. Communication
3. Cryptographic Support
4. Identification and Authentication
5. Security Management
6. Protection of the TSF
7. TOE Access
8. Trusted Path/Channels
9. Intrusion Prevention System

These features are described in more detail in the subsections below.

1.6.1 Security Audit

The TOE provides extensive auditing capabilities. The TOE can audit events related to cryptographic functionality, identification and authentication, and administrative actions. The TOE generates an audit record for each auditable event. The administrator configures auditable events, performs back-up operations, and manages audit data storage. The TOE provides the administrator with a circular audit trail where the TOE overwrites the oldest audit record with the newest audit record when space is full. Audit logs are backed up over an encrypted channel to an external audit server.

1.6.2 Communication

The TOE allows authorized administrators to control which Sensor is managed by the FMC. This is performed through a registration process over TLS. The administrator can also de-register a Sensor if he or she wish to no longer manage it through the FMC.

1.6.3 Cryptographic Support

The TOE provides cryptography in support of other TOE security functionality. The TOE provides cryptography in support of secure connections using IPsec and TLS, and remote administrative management via SSHv2 and TLS/HTTPS. The cryptographic random bit generators (RBGs) are seeded by an entropy noise source.

1.6.4 Identification and authentication

The TOE performs two types of authentication: device-level authentication of the remote device (e.g. IPsec VPN peers) and user authentication for the authorized administrator of the TOE. Device-level authentication allows the TOE to establish a secure channel with a trusted peer. The secure channel is established only after each device authenticates the other. Device-level authentication is performed via IKE/IPsec X509v3 certificate-based authentication or pre-shared key methods.

The TOE provides authentication services for administrative users wishing to connect to the TOEs secure CLI and GUI administrator interfaces. The TOE requires authorized administrators to authenticate prior to being granted access to any of the management functionality. The TOE can be configured to require a minimum password length of 15 characters as well as mandatory password complexity rules. The TOE also implements a lockout mechanism when the number of unsuccessful authentication attempts exceeds the configured threshold.

The TOE provides administrator authentication against a local user database. Password-based authentication can be performed on the serial console or SSH and HTTPS interfaces. The SSHv2 interface also supports authentication using SSH keys.

1.6.5 Security Management

The TOE provides secure administrative services for management of general TOE configuration and the security functionality provided by the TOE. All TOE administration occurs either through a secure SSHv2 or TLS/HTTPS session, or via a local console connection. Optionally, the FTD component also supports tunneling the SSH connections in IPsec VPN tunnels. Management of all security functions can be performed via the FMC/FMCv component of the TOE, while a subset of management functions can be performed on the FTD component. The TOE provides the ability to securely manage all TOE administrative users; all identification and authentication; all audit functionality of the TOE; all TOE cryptographic functionality and the timestamps maintained by the TOE. The TOE supports an "authorized administrator" role, which equates to any account authenticated to an administrative interface (CLI or GUI, but not VPN), and possessing sufficient privileges to perform security-relevant administrative actions.

When an administrative session is initially established, the TOE displays an administrator-configurable warning banner. This is used to provide any information deemed necessary by the administrator. After a configurable period of inactivity, administrative sessions will be terminated, requiring administrators to re-authenticate.

1.6.6 Protection of the TSF

The TOE protects against interference and tampering by untrusted subjects by implementing identification, authentication, and administrator roles to limit configuration to authorized administrators. The TOE prevents reading of cryptographic keys and passwords.

Additionally, the TOE is not a general-purpose operating system and access to the TOE memory space is restricted to only TOE functions.

The TOE internally maintains the date and time. This date and time are used in the timestamp that is applied to audit records generated by the TOE. Administrators can update the TOE's clock manually via FMC. Additionally, the TOE performs testing to verify correct operation of the appliance itself and that of the cryptographic module. Whenever any system failures occur within the TOE the TOE will cease operation.

1.6.7 TOE Access

When an administrative session is initially established, the TOE displays an administrator-configurable warning banner. This is used to provide any information deemed necessary by the administrator. After a configurable period of inactivity, administrator sessions will be terminated, requiring re-authentication.

1.6.8 Trusted path/Channels

The TOE supports establishing trusted paths between itself and remote administrators using SSHv2 for CLI access on the FTD and FMC and TLS/HTTPS for web UI access on the FMC. The TOE supports use of TLS and/or IPsec for connections with remote syslog servers. The SSH remote administrator communications on the FTD can be tunneled in IPsec.

1.6.9 Intrusion Prevention System

The TOE provides intrusion policies consisting of rules and configurations invoked by the access control policy. The intrusion policies are the last line of defense before the traffic is allowed to its destination. All traffic permitted by the access control policy is then inspected by the designated intrusion policy. Using intrusion rules and other preprocessor settings, these policies inspect traffic for security violations and, in inline deployments, can block or alter malicious traffic.

If the vendor-provided intrusion policies do not fully address the security needs of the organization, custom policies can improve the performance of the system in the environment and can provide a focused view of the malicious traffic and policy violations occurring on the network. By creating and tuning custom policies, the administrators can configure, at a very granular level, how the system processes and inspects the traffic on the network for intrusions.

Using Security Intelligence, the administrators can blacklist—deny traffic to and from—specific IP addresses, URLs, and DNS domain names, before the traffic is subjected to analysis by the access control rules. Optionally, the administrators can use a “monitor-only” setting for Security Intelligence filtering.

1.7 Excluded Functionality

The following functionality is excluded from the evaluation.

Table 9: Excluded Functionality

Excluded Functionality	Exclusion Rationale
Telnet for management purposes	Telnet passes authentication credentials in clear text and is disabled by default.
Firepower Device Manager (FDM)	Firepower Device Manager is a web-based local manager. Use of FDM is beyond the scope of this Common Criteria evaluation.
Filtering of non-IP traffic provided by the EtherType option when configuring	Use of non-IP traffic filtering is beyond the scope of this Common Criteria evaluation.

information flow policies is excluded from the evaluated configuration	
Smart Call Home. The Smart Call Home feature provides personalized, e-mail-based and web-based notification to customers about critical events involving their individual systems.	Use of Smart Call Home is beyond the scope of this Common Criteria evaluation.
Shell Access	The shell access is only allowed for pre-operational installation, configuration, and post-operational maintenance and troubling shooting.
NTP	The TOE does not depend on clock updates from NTP servers.
Timeout Exemption Option	The use of the “Exempt from Browser Session Timeout” setting is not permitted. This allows a user to be exempted from the inactivity timeout feature.
REST API	This feature is not evaluated as part of the evaluation. REST API relies on HTTPS as the underlying communication protocol and can be used to build a management interface. This feature is not tested and is out of scope.
Clustering	This feature is not tested and is out of scope.

These services will be disabled by configuration. The exclusion of this functionality does not affect compliance to collaborative Protection Profile for Network Devices (cpp_nd_v2.2e), collaborative Protection Profile for Network Devices Extended Package (EP) for Intrusion Prevention Systems (IPS) (ep_ips_v2.11).

2 CONFORMANCE CLAIMS

2.1 Common Criteria Conformance Claim

The TOE and ST are compliant with the Common Criteria (CC) Version 3.1, Revision 5, dated: April 2017. For a listing of Assurance Requirements claimed see section 5.6.

The TOE and ST are CC Part 2 extended and CC Part 3 conformant.

2.2 Protection Profile Conformance

The TOE and ST are conformant with the Protection Profiles as listed in the table below:

Table 10: Protection Profiles

Protection Profile	Version	Date
collaborative Protection Profile for Network Devices (cpp_nd_v2.2e)	2.2e	23 March 2020
collaborative Protection Profile for Network Devices Extended Package (EP) for Intrusion Prevention Systems (IPS) (ep_ips_v2.11)	2.11	15 June 2017

The TOE and ST are conformant with the Protection Profiles as listed in the table above. The following NIAP Technical Decisions (TD) have also been applied:

Table 11: Technical Decisions

TD #	TD Name	Protection Profiles	Applied to this TOE
TD0592	NIT Technical Decision for Local Storage of Audit Records	CPP_ND_V2.2E	Section A.2.1 of PP
TD0591	NIT Technical Decision for Virtual TOEs and hypervisors	CPP_ND_V2.2E	Assumption – A.LIMITED_FUNCTIONALITY
TD0581	NIT Technical Decision for Elliptic curve-based key establishment and NIST SP 800-56Arev3	CPP_ND_V2.2E	FCS_CKM.2
TD0580	NIT Technical Decision for clarification about use of DH14 in NDcPPv2.2e	CPP_ND_V2.2E	FCS_CKM.1.1, FCS_CKM.2.1
TD0572	NiT Technical Decision for Restricting FTP_ITC.1 to only IP address identifiers	CPP_ND_V2.2E	FTP_ITC.1
TD0571	NiT Technical Decision for Guidance on how to handle FIA_AFL.1	CPP_ND_V2.2E	FIA_AFL.1
TD0570	NiT Technical Decision for Clarification about FIA_AFL.1	CPP_ND_V2.2E	FIA_AFL.1

TD0569	NIT Technical Decision for Session ID Usage Conflict in FCS_DTLSS_EXT.1.7	CPP_ND_V2.2E	FCS_TLSS_EXT.1
TD0564	NiT Technical Decision for Vulnerability Analysis Search Criteria	CPP_ND_V2.2E	AVA_VAN.1
TD0563	NiT Technical Decision for Clarification of audit date information	CPP_ND_V2.2E	FAU_GEN.1
TD0556	NIT Technical Decision for RFC 5077 question	CPP_ND_V2.2E	FCS_TLSS_EXT.1.4, Test 3
TD0555	NIT Technical Decision for RFC Reference incorrect in TLSS Test	CPP_ND_V2.2E	FCS_TLSS_EXT.1.4, Test 3
TD0547	NIT Technical Decision for Clarification on developer disclosure of AVA_VAN	CPP_ND_V2.2E	AVA_VAN.1
TD0546	*NIT Technical Decision for DTLS - clarification of Application Note 63*	*CPP_ND_V2.2E*	*Not applied because this ST does not include FCS_DTLSC_EXT.1.1*
TD0538	NIT Technical Decision for Outdated link to allowed-with list	CPP_ND_V2.2E	Section 2 of PP
TD0537	NIT Technical Decision for Incorrect reference to FCS_TLSC_EXT.2.3	CPP_ND_V2.2E	FCS_TLSC_EXT.2.3
TD0536	NIT Technical Decision for Update Verification Inconsistency	CPP_ND_V2.2E	AGD_OPE.1
TD0528	*NIT Technical Decision for Missing EAs for FCS_NTP_EXT.1.4*	*CPP_ND_V2.2E*	*Not applied because this ST does not include FCS_NTP_EXT.1*
TD0527	Updates to Certificate Revocation Testing (FIA_X509_EXT.1)	CPP_ND_V2.2E	FIA_X509_EXT.1/REV, FIA_X509_EXT.1/ITT
TD0325	Inline mode for Signature-based IPS policies	EP_IPS_V2.11	IPS_SBD_EXT.1.5

2.3 Protection Profile Conformance Claim Rationale

2.3.1 TOE Appropriateness

The TOE provides all of the functionality at a level of security commensurate with that identified in the:

- collaborative Protection Profile for Network Devices (cpp_nd_v2.2e); and
- IPS Extended Package (ep_ips_v2.11)

2.3.2 TOE Security Problem Definition Consistency

The Assumptions, Threats, and Organization Security Policies included in the Security Target represent the Assumptions, Threats, and Organization Security Policies specified in the cpp_nd_v2.2e and ep_ips_v2.11 for which conformance is claimed verbatim. All concepts covered in the Protection Profile Security Problem Definition are included in the Security Target Statement of Security Objectives Consistency.

The Security Objectives included in the Security Target represent the Security Objectives specified in the U.S. Government Protection Profile for Security Requirements for Network Devices for which conformance is claimed verbatim. All concepts covered in the Protection Profile’s Statement of Security Objectives are included in the Security Target.

2.3.3 Statement of Security Requirements Consistency

The Security Functional Requirements included in the Security Target represent the Security Functional Requirements specified in cpp_nd_v2.2e and ep_ips_v2.11 for which conformance is claimed verbatim and several additional Security Functional Requirements are included as a result. All concepts covered the Protection Profile’s Statement of Security Requirements are included in the Security Target. Additionally, the Security Assurance Requirements included in the Security Target are identical to the Security Assurance Requirements included in section 7 of the cpp_nd_v2.2e.

3 SECURITY PROBLEM DEFINITION

This chapter identifies the following:

- Significant assumptions about the TOE's operational environment.
- IT related threats to the organization countered by the TOE.
- Environmental threats requiring controls to provide sufficient protection.
- Organizational security policies for the TOE as appropriate.

This document identifies assumptions as A.assumption with "assumption" specifying a unique name. Threats are identified as T.threat with "threat" specifying a unique name. Organizational Security Policies (OSPs) are identified as P.osp with "osp" specifying a unique name.

3.1 Assumptions

The specific conditions listed in the following subsections are assumed to exist in the TOE's environment. These assumptions include both practical realities in the development of the TOE security requirements and the essential environmental conditions on the use of the TOE.

Table 12: TOE Assumptions

Assumption	Assumption Definition
Reproduced from cpp_nd_v2.2e	
A.PHYSICAL_PROTECTION	The Network Device is assumed to be physically protected in its operational environment and not subject to physical attacks that compromise the security and/or interfere with the device's physical interconnections and correct operation. This protection is assumed to be sufficient to protect the device and the data it contains. As a result, the cPP will not include any requirements on physical tamper protection or other physical attack mitigations. The cPP will not expect the product to defend against physical access to the device that allows unauthorized entities to extract data, bypass other controls, or otherwise manipulate the device. For vNDs, this assumption applies to the physical platform on which the VM runs.
A.LIMITED_FUNCTIONALITY	The device is assumed to provide networking functionality as its core function and not provide functionality/services that could be deemed as general purpose computing. For example, the device should not provide a computing platform for general purpose applications (unrelated to networking functionality). If a virtual TOE evaluated as a pND, following Case 2 vNDs as specified in Section 1.2, the VS is considered part of the TOE with only one vND instance for each physical hardware platform. The exception being where components of a distributed TOE run inside more than one

Assumption	Assumption Definition
	virtual machine (VM) on a single VS. In Case 2 vND, no non-TOE guest VMs are allowed on the platform.
A.NO_THRU_TRAFFIC_PROTECTION	A standard/generic Network Device does not provide any assurance regarding the protection of traffic that traverses it. The intent is for the Network Device to protect data that originates on or is destined to the device itself, to include administrative data and audit data. Traffic that is traversing the Network Device, destined for another network entity, is not covered by the ND cPP. It is assumed that this protection will be covered by cPPs and PP modules for particular types of Network Devices (e.g., firewall).
A.TRUSTED_ADMINSTRATOR	The Security Administrator(s) for the Network Device are assumed to be trusted and to act in the best interest of security for the organization. This includes being appropriately trained, following policy, and adhering to guidance documentation. Administrators are trusted to ensure passwords/credentials have sufficient strength and entropy and to lack malicious intent when administering the device. The Network Device is not expected to be capable of defending against a malicious Administrator that actively works to bypass or compromise the security of the device. For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are expected to fully validate (e.g. offline verification) any CA certificate (root CA certificate or intermediate CA certificate) loaded into the TOE’s trust store (aka 'root store', ' trusted CA Key Store', or similar) as a trust anchor prior to use (e.g. offline verification).
A.REGULAR_UPDATES	The Network Device firmware and software is assumed to be updated by an Administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
A.ADMIN_CREDENTIALS_ SECURE	The Administrator’s credentials (private key) used to access the Network Device are protected by the platform on which they reside.
A.COMPONENTS_RUNNING	For distributed TOEs it is assumed that the availability of all TOE components is checked as appropriate to reduce the risk of an undetected attack on (or failure of) one or more TOE components. It is also assumed that in addition to the availability of all components it is also checked as

Assumption	Assumption Definition
	appropriate that the audit functionality is running properly on all TOE components.
A.RESIDUAL_INFORMATION	The Administrator must ensure that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment.
A.VS_TRUSTED_ADMINISTRATOR	The Security Administrators for the VS are assumed to be trusted and to act in the best interest of security for the organization. This includes not interfering with the correct operation of the device. The Network Device is not expected to be capable of defending against a malicious VS Administrator that actively works to bypass or compromise the security of the device.
A.VS_REGULAR_UPDATES	The VS software is assumed to be updated by the VS Administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
A.VS_ISOLATON	For vNDs, it is assumed that the VS provides, and is configured to provide sufficient isolation between software running in VMs on the same physical platform. Furthermore, it is assumed that the VS adequately protects itself from software running inside VMs on the same physical platform.
A.VS_CORRECT_CONFIGURATION	For vNDs, it is assumed that the VS and VMs are correctly configured to support ND functionality implemented in VMs.
Reproduced from ep_ips_v2.11	
A.CONNECTIONS	It is assumed that the TOE is connected to distinct networks in a manner that ensures that the TOE security policies will be enforced on all applicable network traffic flowing among the attached networks.

3.2 Threats

The following table lists the threats addressed by the TOE and the IT Environment. The assumed level of expertise of the attacker for all the threats identified below is Enhanced-Basic.

Table 13: Threats

Threat	Threat Definition
Reproduced from cpp_nd_v2.2e	
T.UNAUTHORIZED_ ADMINISTRATOR_ACCESS	Threat agents may attempt to gain Administrator access to the Network Device by nefarious means such as masquerading as an Administrator to the device, masquerading as the device to an Administrator, replaying an administrative session (in its entirety, or selected portions), or performing man-in-the-middle attacks, which would provide access to the administrative session, or sessions between Network Devices. Successfully gaining Administrator access allows malicious actions that compromise the security functionality of the device and the network on which it resides.
T.WEAK_CRYPTOGRAPHY	Threat agents may exploit weak cryptographic algorithms or perform a cryptographic exhaust against the key space. Poorly chosen encryption algorithms, modes, and key sizes will allow attackers to compromise the algorithms, or brute force exhaust the key space and give them unauthorized access allowing them to read, manipulate and/or control the traffic with minimal effort.
T.UNTRUSTED_COMMUNICATIONS _CHANNELS	Threat agents may attempt to target Network Devices that do not use standardized secure tunnelling protocols to protect the critical network traffic. Attackers may take advantage of poorly designed protocols or poor key management to successfully perform man-in-the-middle attacks, replay attacks, etc. Successful attacks will result in loss of confidentiality and integrity of the critical network traffic, and potentially could lead to a compromise of the Network Device itself.
T.WEAK_AUTHENTICATION_ ENDPOINTS	Threat agents may take advantage of secure protocols that use weak methods to authenticate the endpoints, e.g. a shared password that is guessable or transported as plaintext. The consequences are the same as a poorly designed protocol, the attacker could masquerade as the Administrator or another device, and the attacker could insert themselves into the network stream and perform a man-in-the-middle attack. The result is the critical network traffic is exposed and there could be a loss of confidentiality and integrity, and potentially the Network Device itself could be compromised.

Threat	**Threat Definition**
T.UPDATE_COMPROMISE	Threat agents may attempt to provide a compromised update of the software or firmware which undermines the security functionality of the device. Non-validated updates or updates validated using non-secure or weak cryptography leave the update firmware vulnerable to surreptitious alteration.
T.UNDETECTED_ACTIVITY	Threat agents may attempt to access, change, and/or modify the security functionality of the Network Device without Administrator awareness. This could result in the attacker finding an avenue (e.g., misconfiguration, flaw in the product) to compromise the device and the Administrator would have no knowledge that the device has been compromised.
T.SECURITY_FUNCTIONALITY_ COMPROMISE	Threat agents may compromise credentials and device data enabling continued access to the Network Device and its critical data. The compromise of credentials includes replacing existing credentials with an attacker’s credentials, modifying existing credentials, or obtaining the Administrator or device credentials for use by the attacker.
T.PASSWORD_CRACKING	Threat agents may be able to take advantage of weak administrative passwords to gain privileged access to the device. Having privileged access to the device provides the attacker unfettered access to the network traffic and may allow them to take advantage of any trust relationships with other Network Devices.
T.SECURITY_FUNCTIONALITY_ FAILURE	An external, unauthorized entity could make use of failed or compromised security functionality and might therefore subsequently use or abuse security functions without prior authentication to access, change or modify device data, critical network traffic or security functionality of the device.
Reproduced from ep_ips_v2.11	
T.NETWORK_DISCLOSURE	Sensitive information on a protected network might be disclosed resulting from ingress-or egress-based actions.
T.NETWORK_ACCESS	Unauthorized access may be achieved to services on a protected network from outside that network, or alternately services outside a protected network from inside the protected network. If malicious external devices are able to communicate with devices on the protected network via a backdoor then those devices may be susceptible to the unauthorized disclosure of information.

Threat	Threat Definition
T.NETWORK_MISUSE	Access to services made available by a protected network might be used counter to Operational Environment policies. Devices located outside the protected network may attempt to conduct inappropriate activities while communicating with allowed public services. E.g. manipulation of resident tools, SQL injection, phishing, forced resets, malicious zip files, disguised executables, privilege escalation tools and botnets.
T.NETWORK_DOS	Attacks against services inside a protected network, or indirectly by virtue of access to malicious agents from within a protected network, might lead to denial of services otherwise available within a protected network. Resource exhaustion may occur in the event of co-ordinate service request flooding from a small number of sources.

3.3 Organizational Security Policies

The following table lists the Organizational Security Policies imposed by an organization to address its security needs.

Table 14: Organizational Security Policies

Policy Name	Policy Definition
Reproduced from cpp_nd_v2.2e	
P.ACCESS_BANNER	The TOE shall display an initial banner describing restrictions of use, legal agreements, or any other appropriate information to which users consent by accessing the TOE.
Reproduced from ep_ips_v2.11	
P.ANALYZE	Analytical processes and information to derive conclusions about potential intrusions must be applied to IPS data and appropriate response actions taken.

4 SECURITY OBJECTIVES

This section identifies the security objectives of the TOE and the IT Environment. The security objectives identify the responsibilities of the TOE and the TOE's IT environment in meeting the security needs.

- ♦ This document identifies objectives of the TOE as O.objective with objective specifying a unique name. Objectives that apply to the IT environment are designated as OE.objective with objective specifying a unique name.

4.1 Security Objectives for the TOE

The following table, Security Objectives for the TOE, identifies the security objectives of the TOE. These security objectives reflect the stated intent to counter identified threats and/or comply with any security policies identified. An explanation of the relationship between the objectives and the threats/policies is provided in the rationale section of this document.

Table 15: Security Objectives for the TOE

TOE Objective	TOE Security Objective Definition
Reproduced from ep_ips_v2.11	
O.SYSTEM_MONITORING	The IPS must collect and store information about all events that may indicate an IPS policy violation related to misuse, inappropriate access, or malicious activity on monitored networks.
O.IPS_ANALYZE	The IPS must apply analytical processes to network traffic data collected from monitored networks and derive conclusions about potential intrusions or network traffic policy violations.
O.IPS_REACT	The IPS must respond appropriately to its analytical conclusions about IPS policy violations.
O.TOE_ADMINISTRATION	The IPS will provide a method for authorized administrator to configure the TSF.
O.TRUSTED_COMMUNICATIONS	The IPS will ensure that communications between distributed components of the TOE are not subject to unauthorized modification or disclosure.

4.2 Security Objectives for the Environment

All of the assumptions stated in section 3.1 are considered to be security objectives for the environment. The following are the Protection Profile non-IT security objectives, which, in addition to those assumptions, are to be satisfied without imposing technical requirements on the

TOE. That is, they will not require the implementation of functions in the TOE hardware and/or software. Thus, they will be satisfied largely through application of procedural or administrative measures.

Table 16: Security Objectives for the Environment

Environment Security Objective	IT Environment Security Objective Definition
Reproduced from cpp_nd_v2.2e	
OE.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.
OE.NO_GENERAL_PURPOSE	There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE. Note: For vNDs the TOE includes only the contents of the its own VM, and does not include other VMs or the VS.
OE.NO_THRU_TRAFFIC_PROTECTION	The TOE does not provide any protection of traffic that traverses it. It is assumed that protection of this traffic will be covered by other security and assurance measures in the operational environment.
OE.TRUSTED_ADMIN	Security Administrators are trusted to follow and apply all guidance documentation in a trusted manner. For vNDs, this includes the VS Administrator responsible for configuring the VMs that implement ND functionality. For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are assumed to monitor the revocation status of all certificates in the TOE's trust store and to remove any certificate from the TOE’s trust store in case such certificate can no longer be trusted.
OE.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.COMPONENTS_RUNNING	For distributed TOEs the Security Administrator ensures that the availability of every TOE component is checked as appropriate to reduce the risk of an undetected attack on (or failure of) one or more TOE components. The Security Administrator also ensures that it is checked as

Environment Security Objective	**IT Environment Security Objective Definition**
	appropriate for every TOE component that the audit functionality is running properly.
OE.RESIDUAL_INFORMATION	The Security Administrator ensures that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment. For vNDs, this applies when the physical platform on which the VM runs is removed from its operational environment.
OE.VM_CONFIGURATION	For vNDs, the Security Administrator ensures that the VS and VMs are configured to • reduce the attack surface of VMs as much as possible while supporting ND functionality (e.g., remove unnecessary virtual hardware, turn off unused inter-VM communications mechanisms), and • correctly implement ND functionality (e.g., ensure virtual networking is properly configured to support network traffic, management channels, and audit reporting). The VS should be operated in a manner that reduces the likelihood that vND operations are adversely affected by virtualization features such as cloning, save/restore, suspend/resume, and live migration. If possible, the VS should be configured to make use of features that leverage the VS's privileged position to provide additional security functionality. Such features could include malware detection through VM introspection, measured VM boot, or VM snapshot for forensic analysis.
Reproduced from ep_ips_v2.11	
OE.CONNECTIONS	TOE administrators will ensure that the TOE is installed in a manner that will allow the TOE to effectively enforce its policies on network traffic of monitored networks.

5 SECURITY REQUIREMENTS

This section identifies the Security Functional Requirements for the TOE. The Security Functional Requirements included in this section are derived from Part 2 of the *Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5, dated: April 2017* and all international interpretations.

5.1 Conventions

The CC defines operations on Security Functional Requirements: assignments, selections, assignments within selections and refinements. This document uses the following font conventions to identify the operations defined by the CC:

- Assignment: Indicated with *italicized* text;
- Refinement made by PP author: Indicated with **bold** text;
- Selection: Indicated with <u>underlined</u> text;
- Iteration: Indicated by appending the iteration number in parenthesis, e.g., (1), (2), (3).
- Where operations were completed in the cpp_nd_v2.2e and ep_ips_v2.11 itself, the formatting used there has been retained.

Extended SFRs are identified by having a label 'EXT' after the requirement name for TOE SFRs. Formatting conventions outside of operations and iterations matches the formatting specified within the PP and EP themselves. In addition, SFRs copied from ep_ips_v2.11 will have extension [IPS] to distinguish them from the cpp_nd_v2.2e. These SFRs that have an extension of [IPS] do not exist in cpp_nd_v2.2e. Changes have been made to the base cPP SFRs as necessary to support the IPS functionality based on ep_ips_v2.11.

Except where noted, all aspects of SFRs are applicable to entire TOE (FTD and FMC). Where specific functionality is only implemented in either FTD or FMC, the applicable subcomponent is identified in an application note, or in embedded qualifiers within the text of the SFR. Application notes clarify distinctions where the TOE includes multiple implementations of a functionality and those implementations differ in their minimum support of the functionality. Thus, the SFR is stating the combined functionality of the TOE.

5.2 TOE Security Functional Requirements

This section identifies the Security Functional Requirements for the TOE. The TOE Security Functional Requirements that appear in the following table are described in more detail in the following subsections.

Table 17: Security Functional Requirements

Class Name	Component Identification	Component Name
Reproduced from cpp_nd_v2.2e		
FAU: Security Audit	FAU_GEN.1	Audit Data Generation
	FAU_GEN.2	User identity association

Class Name	Component Identification	Component Name
	FAU_GEN_EXT.1	Security Audit Generation
	FAU_STG_EXT.1	Protected Audit Event Storage
	FAU_STG_EXT.4	Protected Local Audit Event Storage for Distributed TOEs
	FAU_STG_EXT.5	Protected Remote Audit Event Storage for Distributed TOEs
FCO: Communication	FCO_CPC_EXT.1	Component Registration Channel Definition
FCS: Cryptographic Support	FCS_CKM.1	Cryptographic Key Generation
	FCS_CKM.2	Cryptographic Key Establishment
	FCS_CKM.4	Cryptographic Key Destruction
	FCS_COP.1/DataEncryption	Cryptographic Operation (AES Data Encryption/Decryption)
	FCS_COP.1/SigGen	Cryptographic Operation (Signature Generation and Verification)
	FCS_COP.1/Hash	Cryptographic Operation (Hash Algorithm)
	FCS_COP.1/KeyedHash	Cryptographic Operation (Keyed Hash Algorithm)
	FCS_HTTPS_EXT.1	HTTPS Protocol
	FCS_IPSEC_EXT.1	IPsec Protocol
	FCS_RBG_EXT.1	Random Bit Generation
	FCS_SSHS_EXT.1	SSH Server Protocol
	FCS_TLSC_EXT.1	TLS Client Protocol Without Mutual Authentication
	FCS_TLSC_EXT.2	TLS Client Support for Mutual Authentication
	FCS_TLSS_EXT.1	TLS Server Protocol
FIA: Identification and Authentication	FIA_AFL.1	Authentication Failure Management
	FIA_PMG_EXT.1	Password Management

Class Name	Component Identification	Component Name
	FIA_UIA_EXT.1	User Identification and Authentication
	FIA_UAU_EXT.2	Password-based Authentication Mechanism
	FIA_UAU.7	Protected Authentication Feedback
	FIA_X509_EXT.1/ITT	X.509 Certificate Validation
	FIA_X509_EXT.1/Rev	X.509 Certificate Validation
	FIA_X509_EXT.2(1)	X.509 Certificate Authentication [TLS Clients in FTD for audit]
	FIA_X509_EXT.2(2)	X.509 Certificate Authentication
	FIA_X509_EXT.3	X.509 Certificate Requests
FMT: Security Management	FMT_MOF.1/ManualUpdate	Management of Security Functions Behaviour
	FMT_MTD.1/CoreData	Management of TSF Data
	FMT_MTD.1/CryptoKeys	Management of TSF Data
	FMT_SMF.1	Specification of Management Functions
	FMT_SMR.2	Restrictions on Security Roles
FPT: Protection of the TSF	FPT_SKP_EXT.1	Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)
	FPT_APW_EXT.1	Protection of Administrator Passwords
	FPT_STM_EXT.1	Reliable Time Stamps
	FPT_TST_EXT.1	TSF Testing
	FPT_TUD_EXT.1	Trusted Update
	FPT_ITT.1	Basic internal TSF data transfer protection
	FPT_ITT.1/Join	Basic internal TSF date transfer protection – Registration Channel

Class Name	**Component Identification**	**Component Name**
FTA: TOE Access	FTA_SSL_EXT.1	TSF-initiated Session Locking
	FTA_SSL.3	TSF-initiated Termination
	FTA_SSL.4	User-initiated Termination
	FTA_TAB.1	Default TOE Access Banners
FTP: Trusted path/channels	FTP_ITC.1	Inter-TSF Trusted Channel
	FTP_TRP.1/Admin	Trusted Path
Reproduced from ep_ips_v2.11		
FAU: Security Audit	FAU_GEN.1/IPS[IPS]	Audit Data Generation (IPS)
	FAU_SAR.1[IPS]	Audit Review (IPS Data)
	FAU_SAR.2[IPS]	Restricted Audit Review (IPS Data)
	FAU_SAR.3[IPS]	Selectable Audit Review (IPS Data)
	FAU_STG.1[IPS]	Protected Audit Trail Storage (IPS Data)
FMT: Security Management	FMT_SMF.1/IPS[IPS]	Specification of Management Functions (IPS)
	FMT_MOF.1/IPS[IPS]	Management of Security Functions Behavior
	FMT_MTD.1/IPS[IPS]	Management of IPS Data
	FMT_SMR.2/IPS[IPS]	Security Roles (IPS)
IPS: Intrusion Prevention	IPS_ABD_EXT.1[IPS]	Anomaly-Based IPS Functionality
	IPS_IPB_EXT.1[IPS]	IP Blocking
	IPS_NTA_EXT.1[IPS]	Network Traffic Analysis
	IPS SBD_EXT.1[IPS]	Signature-Based IPS Functionality
FPT: Protection of the TSF	FPT_ITT.1[IPS]	Basic Internal TSF Data Transfer Protection

5.3 SFRs Drawn from CPP_ND_V2.2E

5.3.1 Security audit (FAU)

5.3.1.1 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).*
- *Changes to TSF data related to configuration changes (in addition to the information that a change occurred it shall be logged what has been changed).*
- *Generating/import of, changing, or deleting of cryptographic keys (in addition to the action itself a unique key name or key reference shall be logged).*
- *Resetting passwords (name of related user account shall be logged).*
- *[no other actions];*

d) Specifically defined auditable events listed in Table 18.

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

a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event; and

b) For each audit event type, based on the auditable event definitions of the functional components included in the cPP/ST, *information specified in column three of Table 18.*

Table 18: Auditable Events

SFR	Auditable Event	Additional Audit Record Contents
Reproduced from CPP_ND_V2.2E		
FAU_GEN.1	None.	None.
FAU_GEN.2	None.	None.
FAU_GEN_EXT.1	None.	None.
FAU_STG_EXT.1	None.	None.
FAU_STG_EXT.4	None.	None.
FAU_STG_EXT.5	None.	None.
FCO_CPC_EXT.1	• Enabling communications between a pair of components.	Identities of the endpoint pairs enabled or disabled.

SFR	**Auditable Event**	**Additional Audit Record Contents**
	- Disabling communications between a pair of components.	
FCS_CKM.1	None.	None.
FCS_CKM.2	None.	None.
FCS_CKM.4	None.	None.
FCS_COP.1/ DataEncryption	None.	None.
FCS_COP.1/SigGen	None.	None.
FCS_COP.1/Hash	None.	None.
FCS_COP.1/KeyedHash	None.	None.
FCS_HTTPS_EXT.1	Failure to establish a HTTPS session.	Reason for failure
FCS_IPSEC_EXT.1	Failure to establish an IPsec SA.	Reason for failure.
FCS_RBG_EXT.1	None.	
FCS_SSHS_EXT.1	Failure to establish an SSH session	Reason for failure
FCS_TLSC_EXT.1	Failure to establish a TLS Session	Reason for failure
FCS_TLSC_EXT.2	None.	None.
FCS_TLSS_EXT.1	Failure to establish a TLS Session	Reason for failure
FIA_AFL.1	Unsuccessful login attempts limit is met or exceeded.	Origin of the attempt (e.g., IP address).
FIA_PMG_EXT.1	None.	None.
FIA_UIA_EXT.1	All use of the identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UAU_EXT.2	All use of the identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UAU.7	None.	None.
FIA_X509_EXT.1/ITT	- Unsuccessful attempt to validate a certificate - Any addition, replacement or removal of trust anchors in the TOE's trust store	- Reason for failure of certificate validation - Identification of certificates added, replaced or removed as trust anchor in the TOE's trust store
FIA_X509_EXT.1/Rev	- Unsuccessful attempt to validate a certificate - Any addition, replacement or removal of trust anchors in the TOE's trust store	- Reason for failure of certificate validation - Identification of certificates added, replaced or removed as trust anchor in the TOE's trust store

SFR	Auditable Event	Additional Audit Record Contents
FIA_X509_EXT.2(1)	None.	None.
FIA_X509_EXT.2(2)	None.	None.
FIA_X509_EXT.3	None.	None.
FMT_MOF.1/ManualUpdate	Any attempt to initiate a manual update	None.
FMT_MOF.1/Services	None.	None.
FMT_MTD.1/CoreData	None.	None.
FMT_MTD.1/CryptoKeys	None.	None.
FMT_SMF.1	All management activities of TSF data.	None.
FMT_SMR.2	None.	None.
FPT_SKP_EXT.1	None.	None.
FPT_APW_EXT.1	None.	None.
FPT_TST_EXT.1	None.	None.
FPT_ITT.1	• Initiation of the trusted channel. • Termination of the trusted channel. Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempt
FPT_ITT.1/Join	• Initiation of the trusted channel. • Termination of the trusted channel. Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempt
FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure)	No additional information.
FPT_STM_EXT.1	Discontinuous changes to time - either Administrator actuated or changed via an automated process. (Note that no continuous changes to time need to be logged. See also application note on FPT_STM_EXT.1)	For discontinuous changes to time: The old and new values for the time. Origin of the attempt to change time for success and failure (e.g., IP address).
FTA_SSL_EXT.1	The termination of a local session by the session locking mechanism.	None.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	None.
FTA_SSL.4	The termination of an interactive session.	None.
FTA_TAB.1	None.	None.

SFR	Auditable Event	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/Admin	Initiation of the trusted path. Termination of the trusted path. Failures of the trusted path functions.	None.

5.3.1.2 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.

5.3.1.3 FAU_GEN_EXT.1 Security Audit Generation

FAU_GEN_EXT.1.1 The TSF shall be able to generate audit records for each TOE component. The audit records generated by the TSF of each TOE component shall include the subset of security relevant audit events which can occur on the TOE component.

5.3.1.4 FAU_STG_EXT.1 Protected Audit Event Storage

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

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

- *The TOE shall be a distributed TOE that stores audit data on the following TOE components: [FMC, FTD]*,
- *The TOE shall be a distributed TOE with storage of audit data provided externally for the following TOE components: [FTD transmits its audit data (IPS events) to FMC]*.]

FAU_STG_EXT.1.3 The TSF shall [*overwrite previous audit records according to the following rule: [the newest audit record will overwrite the oldest audit record]*] when the local storage space for audit data is full.

5.3.1.5 FAU_STG_EXT.4 Protected Local Audit Event Storage for Distributed TOEs

FAU_STG_EXT.4.1 The TSF of each TOE component which stores security audit data locally shall perform the following actions when the local storage space for audit data is full:

[

FMC: overwrite previous audit records according to the following rule: [oldest records are overwritten]

FTD: overwrite previous audit records according to the following rule: [oldest records are overwritten]

]

5.3.1.1 FAU_STG_EXT.5 Protected Remote Audit Event Storage for Distributed TOEs

FAU_STG_EXT.5.1 Each TOE component which does not store security audit data locally shall be able to buffer security audit data locally until it has been transferred to another TOE component that stores or forwards it. All transfer of audit records between TOE components shall use a protected channel according to [FPT_ITT.1]

5.3.2 Communication (FCO)

5.3.2.1 FCO_CPC_EXT.1 Communication Partner Control

FCO_CPC_EXT.1.1 The TSF shall require a Security Administrator to enable communications between any pair of TOE components before such communication can take place.

FCO_CPC_EXT.1.2 The TSF shall implement a registration process in which components establish and use a communications channel that uses [

• A channel that meets the secure channel requirements in [FPT_ITT.1/Join]].

for at least TSF data.

FCO_CPC_EXT.1.3 The TSF shall enable a Security Administrator to disable communications between any pair of TOE components.

5.3.3 Cryptographic Support (FCS)

5.3.3.1 FCS_CKM.1 Cryptographic Key Generation

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

- *RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3;*
- *ECC schemes using "NIST curves" [P-256, P-384, P-521] that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4*
- *FFC schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.1*

- *FFC Schemes using ‘safe-prime’ groups that meet the following: “NIST Special Publication 800-56A Revision 3, Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography” and [RFC 3526]*

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

5.3.3.2 FCS_CKM.2 Cryptographic Key Establishment (Refinement)

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

- *RSA-based key establishment schemes that meet the following: RSAES-PKCS1-v1_5 as specified in Section 7.2 of RFC 3447, “Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1”;*

- *Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 3, “Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography”*

- *Finite field-based key establishment schemes that meets the following: NIST Special Publication 800-56A Revision 2, “Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography”*

- *FFC Schemes using “safe-prime” groups that meet the following: ‘NIST Special Publication 800-56A Revision 3, “Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography” and [groups listed in RFC 3526].*

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

5.3.3.3 FCS_CKM.4 Cryptographic Key Destruction

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

- *For plaintext keys in volatile storage, the destruction shall be executed by a [single overwrite consisting of [zeroes], destruction of reference to the key directly followed by a request for garbage collection];*
- *For plaintext keys in non-volatile storage, the destruction shall be executed by the invocation of an interface provided by a part of the TSF that [*
    - *logically addresses the storage location of the key and performs a [[one]-pass] overwrite consisting of [zeroes]];*
    - *instructs a part of the TSF to destroy the abstraction that represents the key]*

that meets the following: *No Standard*.

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

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

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

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

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

]

that meet the following: [

- *For RSA schemes: FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSAPKCS2v1_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*

].

.

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

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

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

FCS_COP.1.1/KeyedHash The TSF shall perform *keyed-hash message authentication* in accordance with a specified cryptographic algorithm *[HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512]* and cryptographic key sizes [*160, 256, 384 and 512 bits*] **and message digest sizes *[160, 256, 384, 512] bits*** that meet the following: *ISO/IEC 9797-2:2011, Section 7 “MAC Algorithm 2”.*

5.3.3.8 FCS_HTTPS_EXT.1 HTTPS Protocol

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

FCS_HTTPS_EXT.1.2 The TSF shall implement HTTPS using TLS.

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

5.3.3.9 FCS_IPSEC_EXT.1 IPsec Protocol

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

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

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

FCS_IPSEC_EXT.1.4 The TSF shall implement the IPsec protocol ESP as defined by RFC 4303 using the cryptographic algorithms [*AES-CBC-128 (RFC 3602), AES-CBC-256 (RFC 3602), AES-GCM-128 (RFC 4106), AES-GCM-256 (RFC 4106)*] together with a Secure Hash Algorithm (SHA)-based HMAC [*HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512*]

FCS_IPSEC_EXT.1.5 The TSF shall implement the protocol: [

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

FCS_IPSEC_EXT.1.7 The TSF shall ensure that [

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

  [

  - *length of time, where the time values can be configured within [120 to 2,147,483,647 seconds. The default is 86,400 seconds which is 24] hours*

  ]

].

FCS_IPSEC_EXT.1.8 The TSF shall ensure that [

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

  [

  - *number of bytes;*

  - *length of time, where the time values can be configured within [120-2,147,483,647 seconds. The default is 28,800 seconds which is 8] hours;*

]

].

FCS_IPSEC_EXT.1.9 The TSF shall generate the secret value x used in the IKE Diffie-Hellman key exchange ("x" in g^x mod p) using the random bit generator specified in FCS_RBG_EXT.1, and having a length of at least [*512*] bits.

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

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

] .

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

[

- [*14 (2048-bit MODP)] according to RFC 3526;*
- *[19 (256-bit Random ECP), 20 (384-bit Random ECP), 24 (2048-bit MODP with 256-bit POS)] according to RFC 5114.*

].

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

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

FCS_IPSEC_EXT.1.14 The TSF shall only establish a trusted channel if the presented identifier in the received certificate matches the configured reference identifier, where the presented and reference identifiers are of the following fields and types: [*SAN: Fully Qualified Domain Name (FQDN), Distinguished Name (DN)*] and [*no other reference identifier type*].

Application Note

In FCS_IPSEC_EXT.1.7, the IKEv2 SA can be limited by time only. In FCS_IPSEC_EXT.1.8, the IKEv2 Child SA can be limited by time or number of kilobytes. The time is in number of seconds.

In FCS_IPSEC_EXT.1.8, the valid range in kilobytes is 10-2,147,483,647 (10KB to 2TB).

5.3.3.10 FCS_RBG_EXT.1 Random Bit Generation

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

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

5.3.3.11 FCS_SSHS_EXT.1(2) SSH Server Protocol

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

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

FCS_SSHS_EXT.1.3 The TSF shall ensure that, as described in RFC 4253, packets greater than [32768] bytes in an SSH transport connection are dropped.

FCS_SSHS_EXT.1.4 The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms and rejects all other encryption algorithms: [*aes128-cbc, aes256-cbc, AEAD_AES_128_GCM, AEAD_AES_256_GCM*].

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

FCS_SSHS_EXT.1.6 The TSF shall ensure that the SSH transport implementation uses *[hmac-sha1, hmac-sha2-256, hmac-sha2-512, AEAD_AES_128_GCM, AEAD_AES_256_GCM]* as its MAC algorithm(s) and rejects all other MAC algorithm(s).

FCS_SSHS_EXT.1.7 The TSF shall ensure that [*diffie-hellman-group14-sha1*] and [*no other methods*] are the only allowed key exchange methods used for the SSH protocol.

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

5.3.3.12 FCS_TLSC_EXT.1 TLS Client Protocol

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

Relevant to syslog over TLS from FTD TLS Client (No mutual authentication):

- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289 (TLSv1.2, TLSv1.1)*
- *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289 (TLSv1.2, TLSv1.1)*
- *TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 (TLSv1.2 only)*

- *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 (TLSv1.2 only)*

Relevant to FPT_ITT.1 (No mutual authentication):

- *TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268 (TLSv1.2, TLSv1.1)*
- *TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268 (TLSv1.2, TLSv1.1)*
- *TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246 (TLSv1.2, TLSv1.1)*
- *TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246 (TLSv1.2, TLSv1.1)*
- *TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288 (TLSv1.2 only)*
- *TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288 (TLSv1.2 only)*
- *TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268 (TLSv1.2, TLSv1.1)*
- *TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268 (TLSv1.2, TLSv1.1)*
- *TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246 (TLSv1.2, TLSv1.1)*
- *TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246 (TLSv1.2, TLSv1.1)*
- *TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492 (TLSv1.2, TLSv1.1)*
- *TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492 (TLSv1.2, TLSv1.1)*
- *TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 (TLSv1.2 only)*
- *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 (TLSv1.2 only)*
- *TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289 (TLSv1.2, TLSv1.1)*
- *TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289 (TLSv1.2, TLSv1.1)*
- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492 (TLSv1.2, TLSv1.1)*
- *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492 (TLSv1.2, TLSv1.1)*
- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289 (TLSv1.2, TLSv1.1)*
- *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289 (TLSv1.2, TLSv1.1)*
- *TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 (TLSv1.2 only)*
- *TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 (TLSv1.2 only)*

Relevant to syslog over TLS from FTD OS TLS Client (No Mutual authentication):

- *TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246 (TLSv1.2, TLSv1.1)*
- *TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246 (TLSv1.2, TLSv1.1)*
- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289 (TLSv1.2, TLSv1.1)*

- *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289 (TLSv1.2, TLSv1.1)*
- *TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 (TLSv1.2 only)*
- *TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 (TLSv1.2 only)*
- *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 (TLSv1.2 only)*
- *TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 (TLSv1.2 only)*
- *TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289 (TLSv1.2, TLSv1.1)*

Relevant to syslog over TLS from FMC/FMCv (With Mutual authentication):

- *TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246 (TLSv1.2, TLSv1.1)*
- *TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246 (TLSv1.2, TLSv1.1)*
- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289 (TLSv1.2, TLSv1.1)*
- *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289 (TLSv1.2, TLSv1.1)*
- *TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 (TLSv1.2 only)*
- *TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 (TLSv1.2 only)*
- *TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 (TLSv1.2 only)*
- *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 (TLSv1.2 only)*
- *TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289 (TLSv1.2, TLSv1.1)*

]and no other ciphersuites

FCS_TLSC_EXT.1.2 The TSF shall verify that the presented identifier matches [*the reference identifier per RFC 6125 section 6, the identifier per RFC 5280 Appendix A using [id-at-title] and no other attribute types*].

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

- *Not implement any administrator override mechanism*.

]

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

Application Note

*FCS_TLSC_EXT.1 is applicable to two TLS clients that send syslog messages to the syslog server- **FTD TLS client**, that is configured by the FMC and is the main audit system for audits generated by FTD. It sends audit events such as IPsec and login messages to the external syslog server and Mutual authentication is not supported; and the **FTD OS TLS client**, that is configured through the FTD's command line and sends audit events to an external syslog server such as SSH login, console login, etc. and Mutual authentication is not supported.*

The selection of – "the identifier per RFC 5280 Appendix A using [id-at-title]"in FCS_TLSC_EXT.1.2 is only applicable to the TLS connection that is relevant to FPT_ITT.1

5.3.3.13 FCS_TLSC_EXT.2 TLS Client Support for Mutual Authentication

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

Application Note

FCS_TLSC_EXT.2 is applicable to the TLS client in FMC/FMCv that is used for transmission of syslog over TLS.

5.3.3.14 FCS_TLSS_EXT.1 TLS Server Protocol

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

Relevant to FPT_ITT.1:

- *TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268 (TLSv1.2, TLSv1.1)*
- *TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268 (TLSv1.2, TLSv1.1)*
- *TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246 (TLSv1.2 only)*
- *TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246 (TLSv1.2 only)*
- *TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288(TLSv1.2 only)*
- *TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288(TLSv1.2 only)*

Relevant to FTP_TRP.1/Admin (applicable to FMC/FMCv only):

- *TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268 (TLSv1.2 only)*
- *TLS_RSA_WITH_AES_256_CBC_ SHA as defined in RFC 3268 (TLSv1.2 only)*
- *TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289 (TLSv1.2 only)*
- *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289 (TLSv1.2 only)*

] and no other ciphersuites.

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

FCS_TLSS_EXT.1.3 The TSF shall perform key establishment for TLS using [*RSA with with key size [2048 bits], ECDHE curves [secp256r1] (**FMC/FMCv**) and no other curves]*

FCS_TLSS_EXT.1.4 The TSF shall support [*session resumption based on session tickets according to RFC 5077*]

5.3.4 Identification and authentication (FIA)

5.3.4.1 FIA_AFL.1 Authentication Failure Management

FIA_AFL.1.1 The TSF shall detect when an Administrator configurable positive integer within [*1 to 99* ***(FMC)****, 1 to 10* ***(FTD)***] unsuccessful authentication attempts occur related to *Administrators attempting to authenticate remotely using a password*.

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

5.3.4.2 FIA_PMG_EXT.1 Password Management

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

a) Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters: [*“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, “)”, [“ ” ‘ ` (double or single quote/apostrophe), + (plus), - (minus), = (equal), , (comma), . (period), / (forward-slash), \ (back-slash), | (vertical-bar or pipe), : (colon), ; (semi-colon), < > (less-than, greater-than inequality signs), [ ] (square-brackets), { } (braces or curly-brackets),^ (caret), _ (underscore), and ~ (tilde)]*];

b) Minimum password length shall be configurable to between *[8] and [32]* characters

5.3.4.3 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 action on behalf of that administrative user.

5.3.4.4 FIA_UAU_EXT.2 Password-based Authentication Mechanism

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

5.3.4.5 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**.

5.3.4.6 FIA_X509_EXT.1/ITT X.509 Certificate Validation

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

- RFC 5280 certificate validation and certificate path validation supporting a minimum path length of two certificates.
- The certificate path must terminate with a trusted CA certificate designated as a trust anchor.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of the certificate using [*no revocation method*].
- The TSF shall validate the extendedKeyUsage field according to the following rules:
    - *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/ITT The TSF shall only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.

5.3.4.7 FIA_X509_EXT.1/Rev X.509 Certificate Validation

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

- RFC 5280 certificate validation and certificate path validation **supporting a minimum path length of three certificates**.
- The certificate path must terminate with a trusted CA certificate designated as a trust anchor.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.

- The TSF shall validate the revocation status of the certificate using [*the Online Certificate Status Protocol (OCSP) as specified in RFC 6960 (**FTD only**), a 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/Rev The TSF shall only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.

5.3.4.8 FIA_X509_EXT.2 X.509(1) Certificate Authentication [FTD OS TLS Client and FMC]

FIA_X509_EXT.2.1(1) The TSF shall use X.509v3 certificates as defined by RFC 5280 to support authentication for [*TLS*], and [*no additional uses*].

FIA_X509_EXT.2.2(1) When the TSF cannot establish a connection to determine the validity of a certificate, the TSF shall [*accept the certificate*].

5.3.4.9 FIA_X509_EXT.2 X.509(2) Certificate Authentication

FIA_X509_EXT.2.1(2) The TSF shall use X.509v3 certificates as defined by RFC 5280 to support authentication for [*IPsec, TLS*], and [*no additional uses*].

FIA_X509_EXT.2.2(2) When the TSF cannot establish a connection to determine the validity of a certificate, the TSF shall [*not accept the certificate*].

5.3.4.10 FIA_X509_EXT.3 X.509 Certificate Requests

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

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

5.3.5 Security management (FMT)

5.3.5.1 FMT_MOF.1/ManualUpdate Management of Security Functions Behaviour

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

5.3.5.2 FMT_MTD.1/CoreData Management of TSF Data

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

5.3.5.1 FMT_MTD.1/CryptoKeys Management of TSF Data

FMT_MTD.1.1/CryptoKeys The TSF shall restrict the ability to *manage* the *cryptographic keys* to *Security Administrators*.

5.3.5.2 FMT_SMF.1 Specification of Management Functions

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

- *Ability to administer the TOE locally and remotely;*
- *Ability to configure the access banner;*
- *Ability to configure the session inactivity time before session termination or locking;*
- *Ability to update the TOE, and to verify the updates using digital signature capability prior to installing those updates;*
- *Ability to configure the authentication failure parameters for FIA_AFL.1;*

  [

  - *Ability to manage the cryptographic keys;*
  - *Ability to configure the cryptographic functionality;*
  - *Ability to configure the lifetime for IPsec SAs;*
  - *Ability to configure the interaction between TOE components;*
  - *Ability to re-enable an Administrator account;*
  - *Ability to set the time which is used for time-stamps;*
  - *Ability to configure the reference identifier for the peer;*
  - *Ability to import X.509v3 certificates to the TOE's trust store;*

]

5.3.5.3 FMT_SMR.2 Restrictions on Security Roles

FMT_SMR.2.1 The TSF shall maintain the roles:

- *Security Administrator.*

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

FMT_SMR.2.3 The TSF shall ensure that the conditions

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

are satisfied.

5.3.6 Protection of the TSF (FPT)

5.3.6.1 FPT_SKP_EXT.1 Protection of TSF Data (for Reading of All Symmetric Keys)

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

5.3.6.2 FPT_APW_EXT.1 Protection of Administrator Passwords

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

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

5.3.6.3 FPT_STM_EXT.1 Reliable time stamps

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

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

5.3.6.4 FPT_TST_EXT.1 TSF Testing

FPT_TST_EXT.1.1 The TSF shall run a suite of the following self-tests [*<u>during initial start-up (on power on)</u>*] to demonstrate the correct operation of the TSF: [*FIPS 140-2 standard power-up self-tests, and firmware integrity test, Noise source health tests*].

5.3.6.5 FPT_TUD_EXT.1 Trusted Update

FPT_TUD_EXT.1.1 The TSF shall provide *Security Administrators* the ability to query the currently executing version of the TOE firmware/software and [*<u>the most recently installed version of the TOE firmware/software</u>*].

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

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

5.3.6.1 FPT_ITT.1: Basic Internal TOE TSF data transfer protection

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

5.3.6.2 FPT_ITT.1/Join: Basic Internal TOE TSF data transfer protection – Registration Channel

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

5.3.7 TOE Access (FTA)

5.3.7.1 FTA_SSL_EXT.1 TSF-initiated Session Locking

FTA_SSL_EXT.1.1 The TSF shall, for local interactive sessions, [

- *<u>terminate the session</u>*]

after a Security Administrator-specified time period of inactivity.

5.3.7.2 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*.

5.3.7.3 FTA_SSL.4 User-initiated Termination

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

5.3.7.4 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.

5.3.8 Trusted Path/Channels (FTP)

5.3.8.1 FTP_ITC.1 Inter-TSF Trusted Channel

FTP_ITC.1.1 The TSF shall **be capable of using [*<u>IPsec (FTD only), TLS</u>*]**] **to** provide a trusted communication channel between itself and **authorized IT entities supporting the following**

capabilities: audit server, [*no other capabilities*] 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 [

- *Audit server: transmit audit data via syslog over IPsec or TLS;*

].

5.3.8.2 FTP_TRP.1/Admin Trusted Path

FTP_TRP.1.1/Admin The TSF shall **be capable of using** [***SSH, IPsec (FTD only), HTTPS (FMC only), TLS (FMC only)***] **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 communicated data**.

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

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

5.4 SFRs from the EP_IPS_v2.11

5.4.1 Security Audit (FAU)

5.4.1.1 FAU_GEN.1/IPS[IPS] Audit Data Generation (IPS)

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

a) Start-up and shut-down of the **IPS** functions;

b) All **IPS** auditable events for the [not specified] level of audit; and

~~*c) All administrative actions;*~~

d) [*All dissimilar IPS events;*

e) All dissimilar IPS reactions;

f) Totals of similar events occurring within a specified time period; and

g) Totals of similar reactions occurring within a specified time period.

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

a) Date and time of the event, type of event **and/or reaction,** ~~subject identity, and the outcome (success or failure) of the event;~~ and;

b) For each **IPS** audit**able** event type, based on the auditable event definitions of the functional components included in the PP~~/ST~~, [*Specifically defined auditable events listed in Table 19*].

Table 19: Auditable Events

SFR	**Auditable Event**	**Additional Audit Record Contents**
Reproduced from the EP_IPS_V2.11		
FMT_SMF.1/IPS [IPS]	Modification of an IPS policy element.	Identifier or name of the modified IPS policy element (e.g. which signature, baseline, or known-good/known-bad list was modified).
IPS_ABD_EXT.1[IPS]	Inspected traffic matches an anomaly-based IPS policy.	Source and destination IP addresses.
		The content of the header fields that were determined to match the policy.
		TOE interface that received the packet.
		Aspect of the anomaly-based IPS policy rule that triggered the event (e.g. throughput, time of day, frequency, etc.).
		Network-based action by the TOE (e.g. allowed, blocked, sent reset to source IP, sent blocking notification to firewall).
IPS_IPB_EXT.1[IPS]	Inspected traffic matches a list of known-good or known-bad addresses applied to an IPS policy.	Source and destination IP addresses (and, if applicable, indication of whether the source and/or destination address matched the list).
		TOE interface that received the packet.
		Network-based action by the TOE (e.g. allowed, blocked, sent reset).
IPS_NTA_EXT.1[IPS]	Modification of which IPS policies are active on a TOE interface. Enabling/disabling a TOE interface with IPS policies applied. Modification of which mode(s) is/are active on a TOE interface.	Identification of the TOE interface.
		The IPS policy and interface mode (if applicable).
IPS_SBD_EXT.1[IPS]	Inspected traffic matches a signature-based IPS rule with logging enabled.	Name or identifier of the matched signature.
		Source and destination IP addresses.
		The content of the header fields that were determined to match the signature.
		TOE interface that received the packet.
		Network-based action by the TOE (e.g. allowed, blocked, sent reset).

5.4.1.1 FAU_SAR.1[IPS] Audit Review (IPS Data)

FAU_SAR.1.1[IPS] Refinement: The TSF shall provide [*authorized administrators*] with the capability to read [*IPS data*] from the ~~audit records~~ IPS events.

FAU_SAR.1.2[IPS] Refinement: The TSF shall provide the ~~audit records~~ **IPS data** in a manner suitable for the ~~user~~ **administrators** to interpret the information.

5.4.1.1 FAU_SAR.2[IPS] Restricted Audit Review (IPS Data)

FAU_SAR.2.1[IPS] Refinement: The TSF shall prohibit all ~~users~~ **administrators** read access to the ~~audit records~~ **IPS data,** except those that have been granted explicit read-access.

5.4.1.1 FAU_SAR.3[IPS] Selectable Audit Review (IPS Data)

FAU_SAR.3.1[IPS] Refinement: The TSF shall provide the ability to apply [*filtering and sorting*] of ~~audit~~ **IPS data** based on [*filtering parameters: risk rating, time period, source IP address, destination IP address and* [<u>*other filtering parameters described in the TSS*</u>*]*]; *and sorting parameters: event ID, event type, time, signature ID, IPS actions performed, and* [*[*<u>*other sorting parameters described in the TSS*</u>*]*].

5.4.1.1 FAU_STG.1[IPS] Protected Audit Trail Storage (IPS Data)

FAU_STG.1.1[IPS] Refinement: The TSF shall protect the stored ~~audit records~~ **IPS data** from unauthorized deletion.

FAU_STG.1.2[IPS] Refinement: The TSF shall be able to [*prevent*] unauthorized modifications to the stored ~~audit records~~ **IPS data** ~~in the audit trail~~.

5.4.2 Security management (FMT)

5.4.2.1 FMT_SMF.1/IPS[IPS] Specification of Management Functions (IPS)

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

- *Enable, disable signatures applied to sensor interfaces, and determine the behavior of IPS functionality*
- *Modify these parameters that define the network traffic to be collected and analyzed:*
    - *Source IP addresses (host address and network address)*
    - *Destination IP addresses (host address and network address)*
    - *Source port (TCP and UDP)*
    - *Destination port (TCP and UDP)*
    - *Protocol (IPv4 and IPv6)*
    - *ICMP type and code*
- *Update (import) signatures*
- *Create custom signatures*
- *Configure anomaly detection*
- *Enable and disable actions to be taken when signature or anomaly matches are detected*
- *Modify thresholds that trigger IPS reactions*
- *Modify the duration of traffic blocking actions*
- *Modify the known-good and known-bad lists (of IP addresses or address ranges)*
- *Configure the known-good and known-bad lists to override signature-based IPS policies*]

5.4.2.2 FMT_MOF.1/IPS[IPS] Management of Security Functions Behavior*

FMT_MOF.1.1/IPS[IPS] The TSF shall restrict the ability to modify the behavior of the functions [*IPS data collection, analysis, and reaction*] to [*authorized IPS Administrators*].

5.4.2.3 FMT_MTD.1/IPS[IPS] Management of IPS Data*

FMT_MTD.1.1/IPS[IPS] Refinement: The TSF shall restrict the ability to [change_default, query, modify, delete, clear] the [*all **IPS** data*] to [*the IPS Administrator, IPS Analyst and other IPS-specific roles identified in FMT_SMR.2/IPS*].

5.4.2.4 FMT_SMR.2/IPS[IPS] Security Roles (IPS)*

FMT_SMR.2.1/IPS[IPS] Refinement: The TSF shall maintain the roles: [***IPS Administrator, IPS Analyst, and [[Access Admin, Discovery Admin, Security Analyst]]***].

FMT_SMR.2.2/IPS[IPS] The TSF shall be able to associate users with roles.

FMT_SMR.2.3/IPS[IPS] The TSF shall ensure that [

- *IPS Administrator (or Administrator): Have all privileges and access*
- *IPS Analyst (or Intrusion Admin): Have all access to intrusion policies and network analysis privileges but cannot deploy policies*
- *Access Admin: Have all access to access control policies but cannot deploy policies*
- *Discovery Admin: Have all access to network discovery, application detection, and correlation features but cannot deploy policies*
- *Security Analyst: Have all access to security event analysis feature*

] are satisfied.

5.4.3 Intrusion Prevention (IPS)

5.4.3.1 IPS_ABD_EXT.1[IPS] Anomaly-Based IPS Functionality

IPS_ABD_EXT.1.1[IPS] The TSF shall support the definition of [anomaly ('unexpected') traffic patterns] including the specification of [

- frequency;
- *[preprocessor detection rules for anomaly detected in headers and protocols]*]

and the following network protocol fields:

- [all packet header and data elements defined in IPS_SBD_EXT.1]

Application Note

Although the term "threshold" is used in the TSS, the TOE's definition of "threshold" matches the definition of frequency in the ep_ips_v2.11. Therefore, "frequency", rather than "threshold" has been selected in the IPS_ABD_EXT.1.1 requirement.

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

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

- In any mode, for any sensor interface: [
  - allow the traffic flow]
- In inline mode: [
  - allow the traffic flow
  - block/drop the traffic flow

  and [no other actions]

5.4.3.2 IPS_IPB_EXT.1[IPS] IP Blocking

IPS_IPB_EXT.1.1[IPS] The TSF shall support configuration and implementation of known-good and known-bad lists of [source, destination] IP addresses.

IPS_IPB_EXT.1.2[IPS] The TSF shall allow IPS Administrators and [*[Intrusion Admin, Access Admin]*] to configure the following IPS policy elements: [known-good list rules, known-bad list rules, IP addresses, *[Domain names and URLs]*].

5.4.3.3 IPS_NTA_EXT.1[IPS] Network Traffic Analysis

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

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

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

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

- Promiscuous (listen-only) mode: [*Giga Ethernet*];
- Inline (data pass-through) mode: [*Giga Ethernet*];

- Management mode: [*Giga Ethernet*];
- [
    - no other interface types].

5.4.3.4 IPS_SBD_EXT.1[IPS] Signature-Based IPS Functionality

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

- IPv4: Version; Header Length; Packet Length; ID; IP Flags; Fragment Offset; Time to Live (TTL); Protocol; Header Checksum; Source Address; Destination Address; and IP Options and [no other field].
- IPv6: Version; payload length; next header; hop limit; source address; destination address; routing header; and [no other field].
- ICMP: Type; Code; Header Checksum; and [ID, sequence number, *[no other field]*].
- ICMPv6: Type; Code; and Header Checksum.
- TCP: Source port; destination port; sequence number; acknowledgement number; offset; reserved; TCP flags; window; checksum; urgent pointer; and TCP options.
- UDP: Source port; destination port; length; and UDP checksum.

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

- ICMPv4 data: characters beyond the first 4 bytes of the ICMP header.
- ICMPv6 data: characters beyond the first 4 bytes of the ICMP header.
- TCP data (characters beyond the 20 byte TCP header), with support for detection of:

    i) FTP (file transfer) commands: help, noop, stat, syst, user, abort, acct, allo, appe, cdup, cwd, dele, list, mkd, mode, nlst, pass, pasv, port, pass, quit, rein, rest, retr, rmd, rnfr, rnto, site, smnt, stor, stou, stru, and type.

    ii) HTTP (web) commands and content: commands including GET and POST, and administrator-defined strings to match URLs/URIs, and web page content.

    iii) SMTP (email) states: start state, SMTP commands state, mail header state, mail body state, abort state.

    iv) [no other types of TCP payload inspection];

- UDP data: characters beyond the first 8 bytes of the UDP header;
- [*no other types of packet payload inspection*]

In addition, the TSF shall support stream reassembly or equivalent to detect malicious payload even if it is split across multiple non-fragmented packets.

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

a) IP Attacks

i) IP Fragments Overlap (Teardrop attack, Bonk attack, or Boink attack)

ii) IP source address equal to the IP destination (Land attack)

b) ICMP Attacks

i) Fragmented ICMP Traffic (e.g. Nuke attack)

ii) Large ICMP Traffic (Ping of Death attack)

c) TCP Attacks

i) TCP NULL flags

ii) TCP SYN+FIN flags

iii) TCP FIN only flags

iv) TCP SYN+RST flags

d) UDP Attacks

i) UDP Bomb Attack

ii) UDP Chargen DoS Attack

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

a) Flooding a host (DoS attack)

i) ICMP flooding (Smurf attack, and ping flood)

ii) TCP flooding (e.g. SYN flood)

b) Flooding a network (DoS attack)

c) Protocol and port scanning

i) IP protocol scanning

ii) TCP port scanning

iii) UDP port scanning

iv) ICMP scanning

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

- In any mode, for any sensor interface: [
    - o allow the traffic flow;]
- In inline mode:
    - o block/drop the traffic flow;
    - o and [
        - § allow all traffic flow;

]

5.4.4 Protection of the TSF (FPT)

5.4.4.1 FPT_ITT.1[IPS] Basic Internal TSF Data Transfer Protection

FPT_ITT.1.1[IPS] Refinement: The TSF shall protect TSF data from [disclosure, modification] **using [TLS]** when it is transmitted between [*the FMC and Sensor*] ~~separate parts of the TOE.~~

5.5 TOE SFR Dependencies Rationale for SFRs Found in CPP_ND_V2.2E and EP_IPS_V2.11

The CPP_ND_V2.2E and EP_IPS_V2.11 contains all the requirements claimed in this Security Target. As such the dependencies are not applicable since the PP itself has been approved.

5.6 Security Assurance Requirements

5.6.1 SAR Requirements

The TOE assurance requirements for this ST are taken directly from the CPP_ND_V2.2E which are derived from Common Criteria Version 3.1, Revision 5. The assurance requirements are summarized in the table below.

Table 20: Assurance Measures

Assurance Class	Components	Components Description
DEVELOPMENT	ADV_FSP.1	Basic Functional Specification
GUIDANCE DOCUMENTS	AGD_OPE.1	Operational User Guidance
	AGD_PRE.1	Preparative User Guidance
LIFE CYCLE SUPPORT	ALC_CMC.1	Labeling of the TOE
	ALC_CMS.1	TOE CM Coverage
TESTS	ATE_IND.1	Independent Testing - Conformance
VULNERABILITY ASSESSMENT	AVA_VAN.1	Vulnerability Analysis

5.6.2 Security Assurance Requirements Rationale

This Security Target claims conformance to the CPP_ND_V2.2E. This target was chosen to ensure that the TOE has a basic to moderate level of assurance in enforcing its security functions when instantiated in its intended environment which imposes no restrictions on assumed activity on applicable networks. The ST also claims conformance to EP_IPS_V2.11, which includes refinements to assurance measures for the SFRs defined in the two aforementioned modules including augmenting the vulnerability analysis (AVA_VAN.1) with specific vulnerability testing.

5.7 Assurance Measures

The TOE satisfies the identified assurance requirements. This section identifies the Assurance Measures applied by Cisco to satisfy the assurance requirements. The table below lists the details.

Table 21: Assurance Measures

Component	How requirement will be met
ADV_FSP.1	The functional specification describes the external interfaces of the TOE; such as the means for a user to invoke a service and the corresponding response of those services. The description includes the interface(s) that enforces a security functional requirement, the interface(s) that supports the enforcement of a security functional requirement, and the interface(s) that does not enforce any security functional requirements. The interfaces are described in terms of their purpose (general goal of the interface), method of use (how the interface is to be used), parameters (explicit inputs to and outputs from an interface that control the behavior of that interface), parameter descriptions (tells what the parameter is in some meaningful way), and error messages (identifies the condition that generated it, what the message is, and the meaning of any error codes). The development evidence also contains a tracing of the interfaces to the SFRs described in this ST.
AGD_OPE.1	The Administrative Guide provides the descriptions of the processes and procedures of how the administrative users of the TOE can securely administer the TOE using the interfaces that provide the features and functions detailed in the guidance.
AGD_PRE.1	The Installation Guide describes the installation, generation, and startup procedures so that the users of the TOE can put the components of the TOE in the evaluated configuration.
ALC_CMC.1	The Configuration Management (CM) document(s) describes how the consumer (end-user) of the TOE can identify the evaluated TOE (Target of Evaluation). The CM document(s) identifies the configuration items, how those configuration items are uniquely identified, and the adequacy of the procedures that are used to control and track changes that are made to the TOE. This includes details on what changes are tracked, how potential changes are incorporated, and the degree to which automation is used to reduce the scope for error.
ALC_CMS.1	
ATE_IND.1	Cisco provides the TOE for testing.
AVA_VAN.1	Cisco provides the TOE for testing.

6 TOE SUMMARY SPECIFICATION

6.1 TOE Security Functional Requirement Measures

This chapter identifies and describes how the Security Functional Requirements identified above are met by the TOE.

Table 22: How TOE SFRs Are Satisfied

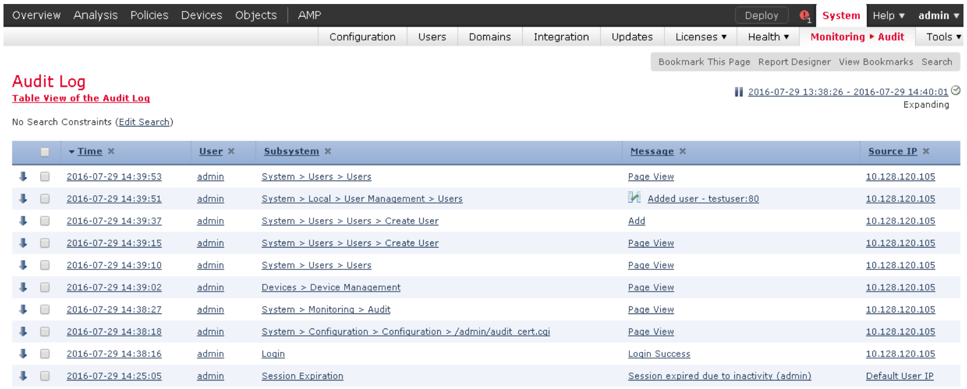
<table>
<tr><th>TOE SFRs</th><th>How the SFR is Satisfied</th></tr>
<tr><td colspan="2"><b>Security Functional Requirements Drawn from CPP_ND_V2.2E</b></td></tr>
<tr><td>FAU_GEN.1,
FAU_GEN_EXT.1,
FAU_STG_EXT.1,
FAU_STG_EXT.4,
FAU_STG_EXT.5</td><td>

Auditing is the recording of events within the system. The TOE generates 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 the audit function[1], any use of an administrator command or action via the CLI and web interfaces, and all of the required auditable events identified in Table 18. For more information about the required audit events, please refer to Table 18 and the operational user guide (also known as the CC Supplemental User guide).

The FMC component of the TOE can generate an audit record for each user interaction with the web interface and each command in the CLI interface. The FTD component of the TOE can generate traffic events as part of the intrusion (IPS) and these event records are stored in logs separate from the audit logs for performance and security reasons. For more details about which auditable events are mapped to which SFRs, refer to Table 27.

FMC and Sensors log auditing information for all user activity in a read-only format. Modifications are not allowed by the interfaces and only authorized administrators can delete the audit logs. Audit logs are presented in a standard event view that allows administrators to view, sort, and filter audit log messages based on any item in the audit view. The audit view contains columns with information field for each audit event such as time, user, subsystem, message, and source IP. Please see the figure below for example.

Figure 6: Audit View



<table>
<tr><th></th><th>□</th><th>▾ Time ✖</th><th>User ✖</th><th>Subsystem ✖</th><th>Message ✖</th><th>Source IP ✖</th></tr>
<tr><td>⬇</td><td>□</td><td>2016-07-29 14:39:53</td><td>admin</td><td>System &gt; Users &gt; Users</td><td>Page View</td><td>10.128.120.105</td></tr>
<tr><td>⬇</td><td>□</td><td>2016-07-29 14:39:51</td><td>admin</td><td>System &gt; Local &gt; User Management &gt; Users</td><td>Added user - testuser:80</td><td>10.128.120.105</td></tr>
<tr><td>⬇</td><td>□</td><td>2016-07-29 14:39:37</td><td>admin</td><td>System &gt; Users &gt; Users &gt; Create User</td><td>Add</td><td>10.128.120.105</td></tr>
<tr><td>⬇</td><td>□</td><td>2016-07-29 14:39:15</td><td>admin</td><td>System &gt; Users &gt; Users &gt; Create User</td><td>Page View</td><td>10.128.120.105</td></tr>
<tr><td>⬇</td><td>□</td><td>2016-07-29 14:39:10</td><td>admin</td><td>System &gt; Users &gt; Users</td><td>Page View</td><td>10.128.120.105</td></tr>
<tr><td>⬇</td><td>□</td><td>2016-07-29 14:39:02</td><td>admin</td><td>Devices &gt; Device Management</td><td>Page View</td><td>10.128.120.105</td></tr>
<tr><td>⬇</td><td>□</td><td>2016-07-29 14:38:27</td><td>admin</td><td>System &gt; Monitoring &gt; Audit</td><td>Page View</td><td>10.128.120.105</td></tr>
<tr><td>⬇</td><td>□</td><td>2016-07-29 14:38:18</td><td>admin</td><td>System &gt; Configuration &gt; Configuration &gt; /admin/audit_cert.cgi</td><td>Page View</td><td>10.128.120.105</td></tr>
<tr><td>⬇</td><td>□</td><td>2016-07-29 14:38:16</td><td>admin</td><td>Login</td><td>Login Success</td><td>10.128.120.105</td></tr>
<tr><td>⬇</td><td>□</td><td>2016-07-29 14:25:05</td><td>admin</td><td>Session Expiration</td><td>Session expired due to inactivity (admin)</td><td>Default User IP</td></tr>
</table>

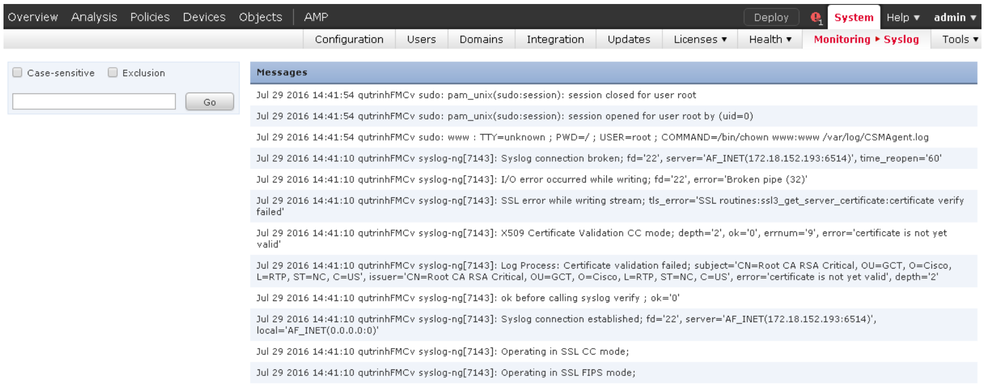
The following fields are recorded for each audit event in the audit view:

- **Time**: The time and date that the appliance generated the audit record.

</td></tr>
</table>

[1] Note that the audit function cannot be disabled other than shutting down the entire system.

TOE SFRs	How the SFR is Satisfied

	• **User**: The user name of the user that triggered the audit event.
	• **Subsystem**: The menu path the user followed to generate the audit record. For example, “System > Monitoring > Audit” is the menu path to view the audit log.
	• **Message**: The action the user performed. For example, “Page View” signifies that the user simply viewed the page indicated in the Subsystem, while “Save” means that the user clicked the Save button on the page.
	• **Source IP**: The IP address of the host used by the user.
	Figure 7: Syslog View
	
	 Overview Analysis Policies Devices Objects AMP Deploy System Help ▾ admin ▾ Configuration Users Domains Integration Updates Licenses ▾ Health ▾ Monitoring ▸ Syslog Tools ▾ □ Case-sensitive □ Exclusion Go
	Messages
	`Jul 29 2016 14:41:54 qutrinhFMCv sudo: pam_unix(sudo:session): session closed for user root`
	`Jul 29 2016 14:41:54 qutrinhFMCv sudo: pam_unix(sudo:session): session opened for user root by (uid=0)`
	`Jul 29 2016 14:41:54 qutrinhFMCv sudo: www : TTY=unknown ; PWD=/ ; USER=root ; COMMAND=/bin/chown www:www /var/log/CSMAgent.log`
	`Jul 29 2016 14:41:10 qutrinhFMCv syslog-ng[7143]: Syslog connection broken; fd='22', server='AF_INET(172.18.152.193:6514)', time_reopen='60'`
	`Jul 29 2016 14:41:10 qutrinhFMCv syslog-ng[7143]: I/O error occurred while writing; fd='22', error='Broken pipe (32)'`
	`Jul 29 2016 14:41:10 qutrinhFMCv syslog-ng[7143]: SSL error while writing stream; tls_error='SSL routines:ssl3_get_server_certificate:certificate verify failed'`
	`Jul 29 2016 14:41:10 qutrinhFMCv syslog-ng[7143]: X509 Certificate Validation CC mode; depth='2', ok='0', errnum='9', error='certificate is not yet valid'`
	`Jul 29 2016 14:41:10 qutrinhFMCv syslog-ng[7143]: Log Process: Certificate validation failed; subject='CN=Root CA RSA Critical, OU=GCT, O=Cisco, L=RTP, ST=NC, C=US', issuer='CN=Root CA RSA Critical, OU=GCT, O=Cisco, L=RTP, ST=NC, C=US', error='certificate is not yet valid', depth='2'`
	`Jul 29 2016 14:41:10 qutrinhFMCv syslog-ng[7143]: ok before calling syslog verify ; ok='0'`
	`Jul 29 2016 14:41:10 qutrinhFMCv syslog-ng[7143]: Syslog connection established; fd='22', server='AF_INET(172.18.152.193:6514)', local='AF_INET(0.0.0.0:0)'`
	`Jul 29 2016 14:41:10 qutrinhFMCv syslog-ng[7143]: Operating in SSL CC mode;`
	`Jul 29 2016 14:41:10 qutrinhFMCv syslog-ng[7143]: Operating in SSL FIPS mode;`
	The user can also view the audit log using the command “show audit-log” or “show syslog” via the CLI interface. All GUI actions and CLI commands are recorded in the audit log and can only be viewed by authorized administrators. To distinguish between the two, the Subsystem field will identify “Command Line” for commands and the Message field will identify the executed command.
	In general, the logged audit records identify the date and time, the identity of the actor (e.g., user, daemon, or network host) responsible for the event, the subsystem that triggers the event, an indication of whether the event succeeded, failed or had some other outcome (if applicable), and the source IP (if applicable). The logged audit records also include event-specific content that includes at least all of the content required in table above.
	The TOE (FMC) includes an internal log database implementation that can be used to store and review audit records locally. However, the internal log only stores a default of 100,000 entries in the local database (to configure the size, go to System > Configuration > Database, and click on “Audit Event Database”). When the audit log is full, the oldest audit records are overwritten by the newest audit records. In addition, the TOE (FMC) also includes a local syslog storage in /var/log/messages and these logs are viewable through the FMC GUI. The contents are stored in flat files which are rotated automatically. Similar to the audit log, when the syslog is full, the oldest syslog messages are overwritten by the newest one.
	For audit log, the events are stored in partitioned event tables. The TOE will prune (i.e., delete) the oldest partition whenever the oldest partition can be pruned without dropping the number of events count below the configured event limit. Note this limit defaults to 10,000 if you set it any lower. For example, if you set the limit to 10,000 events, the events count may need to exceed 15,000 events before the oldest partition can be deleted. For syslog, the logs are stored in /var/log/messages and are rotated daily or when the log file size exceeds 25 MB. After the maximum number of backlog files is reached, the oldest is deleted and the numbers on the other backlogs file are incremented.

<table>
<tr><th>TOE SFRs</th><th>How the SFR is Satisfied</th></tr>
<tr><td></td><td>
To prevent the losing of critical audit records, the administrators can configure the system to transmit all the audit events (i.e., audit log and syslog) in real-time over a secure TLS or an IPsec (FTD only) connection to an external audit server in the operational environment. When an audit event is generated, it is sent to the local storage and external audit server simultaneously. This ensures that current audit events can be viewed locally while all events, new or old, are stored off-line as required by the NDcPP.
Note that the protection of the audit records stored at the external audit server is the responsibility of the operational environment. The TOE is only responsible for the secure communication channel. It is recommended that the audit server is physically or logically separated (e.g., VLANs) from the other networks.
The TOE can be configured to export syslog records to an administrator-specified, external syslog server. The TOE can be configured to encrypt the communications with an external syslog server using IPsec or TLS. FMC transmits syslog over TLS and FTD transmits syslog over TLS and IPsec.
The audit records are also stored locally and when the local storage is full, the newest data will overwrite the oldest data. On FMC, log messages (those generated locally and those forwarded from FTD) are stored locally on FMC in a database. Different message types are stored separately in local databases, and each local store has a separately configurable size limit (configurable in FMC via System > Configuration > Database). Admin actions are stored in the Audit Event Database, IPS events are stored in Intrusion Event Database.
Messages generated by FTD are stored locally but the IPS events generated by FTD are immediately transmitted via secure TLS channel from FTD to FMC for retention in the FMC databases. As messages are generated by FTD, they are immediately transmitted from FTD to a remote syslog server and stored in a local buffer (buffer size configurable from 4096-52428800 bytes) which overwrites old messages with new ones when storage limits are reached. Thee local logs are viewable from the FTD CLI shell by using “show logging”.
For audit messages related to management of cryptographic keys, the audit message details include the name of the certificate associated with the key.
<u>Samples Audit Events</u>
Nov 21 2012 20:39:21: %ASA-3-713194: Group = 192.168.22.1, IP = 192.168.22.1, Sending IKE Delete With Reason message: Disconnected by Administrator.
Network interfaces have bandwidth limitations, and other traffic flow limitations that are configurable. When an interface has exceeded a limit for processing traffic, traffic will be dropped, and audit messages can be generated, such as:
Nov 21 2012 20:39:21: %ASA-3-201011: Connection limit exceeded <i>cnt</i>/<i>limit</i> for <i>dir</i> packet from <i>sip</i>/<i>sport</i> to <i>dip</i>/<i>dport</i> on interface <i>if_name</i>.
Nov 21 2012 20:39:21: %ASA-3-202011: Connection limit exceeded <i>econns/limit</i> for <i>dir</i> packet from <i>source_address/source_port</i> to <i>dest_address/dest_port</i> on interface <i>interface_name</i>
For more information on the required auditable events and the actual logs themselves, please refer to the Preparative Procedures & Operational User Guide for the Common Criteria Certified Configuration.
The following high-level events are auditable by the TOE:
<table>
<tr><th>Auditable Event</th><th>Rationale</th></tr>
<tr><td>Modifications to the group of users that are part of the authorized administrator role.</td><td>All changes to the configuration (and hence all security relevant administrator actions) are logged when the logging level is set to at least the</td></tr>
</table>
</td></tr>
</table>

TOE SFRs	How the SFR is Satisfied	
		'notifications' level. These changes would fall into the category of configuration changes such as enabling or disabling features and services. The identity of the administrator taking the action and the user being affected (assigned to the authorized administrator role) are both included within the event.
	All use of the user identification mechanism.	Events will be generated for attempted identification/ authentication, and the username attempting to authenticate will be recorded in the event.
	Any use of the authentication mechanism.	Events will be generated for attempted identification/ authentication, and the username attempting to authenticate will be recorded in the event along with the origin or source of the attempt.
	The reaching of the threshold for unsuccessful authentication attempts and the subsequent restoration by the authorized administrator of the user's capability to authenticate.	Failed attempts for authentication will be logged, and when the threshold is reached, it will also be logged. All changes to the configuration are logged when the logging level is set to at least the 'notifications' level. Changes to restore a locked account would fall into the category of configuration changes.
	Success and failure, and the type of cryptographic operation	Attempts for VPN connections are logged (whether successful or failed). Requests for encrypted session negotiation are logged (whether successful or failed). The identity of the user performing the cryptographic operation is included in the event.
	Failure to establish and/or establishment/termination of an IPsec session	Attempts to establish an IPsec tunnel or the failure of an established IPsec tunnel is logged as well as successfully established and terminated IPsec sessions with peer.
	Changes to the time.	Changes to the time are logged with old and new time values.
	Use of the functions listed in this requirement pertaining to audit.	All changes to the configuration are logged when the logging level is set to at least the 'notifications' level. These changes would fall into the category of configuration changes.
	Loss of connectivity with an external syslog server.	Loss of connectivity with an external syslog server is logged as a terminated or failed cryptographic channel.
	Initiation of an update to the TOE.	TOE updates are logged as configuration changes.
	Termination of local and remote sessions. Note that the TOE does not support session locking, so	Termination of a local and remote session is logged. This also includes termination of remote VPN session as well. The user may initiate or the system

TOE SFRs	How the SFR is Satisfied	
	there is no corresponding audit.	may terminate the session based idle timeout setting.
	Initiation, termination and failures in trusted channels and paths.	Requests for encrypted session negotiation are logged (whether successful or failed). Similarly, when an established cryptographic channel or path is terminated or fails a log record is generated. This applies to HTTPS, TLS, IPsec, and SSH.
	Successful SSH rekey	SSH rekey event is logged.
	Indication of packets dropped due to too much network traffic	Logs are generated when traffic that exceeds the settings allowed on an interface is received.
FAU_GEN.2	The TOE ensures each action performed by the administrator at the CLI and web GUI is logged with the administrator's identity and as a result events are traceable to a specific user.	
FCO_CPC_EXT.1	In order for TOE components to communicate as part of a distributed TOE System, they must successfully complete a registration process. Each TOE component comes with a manufacture's TLS certificate. To start the registration process, the administrator must enable or register the TOE components. On the FMC, the administrator must go to Device Management UI and click on "Add Device". At the same time, the administrator must go to the Sensor CLI, and click or enter "Configure Manager Add". The administrator must specify the peer hostname or IP address and the registration key used for the initial authentication. During the registration process, the manufacture's TLS certificates are used to setup the initial TLS channel on the internal trusted management network. If the authentication succeeded, the resident CA on the FMC will sign and issue a TLS certificate along with the private key to the Sensor which will be used for subsequent TLS channel. To disable or de-register a Sensor, the administrator must initiate a "Delete Device" on the FMC Device Management UI and then perform a "Configure Manager Delete" action on the CLI of the FTD. This will destroy (i.e., zeroize) the TLS certificate and private key. Once this has occurred, no farther communication can happen without another registration process.	
FCS_CKM.1, FCS_CKM.2, FCS_CKM.4, FCS_COP.1/ DataEncryption, FCS_COP.1/SigGen, FCS_COP.1/Hash, FCS_COP.1/ KeyedHash, and FCS_RBG_EXT.1	Each FMC (including the virtual appliance) and each Sensor "TOE" utilizes a cryptographic module (i.e., Cisco FIPS Object Module) to provide supporting cryptographic functions. When the term "TOE" is used in this section, it refers to each appliance. The algorithm implementations have been tested in accordance to validation suites set by the Cryptographic Algorithm Validation Program (CAVP) and tested on specific processors. Refer to section 7.3 of this document for the listings of CAVP certificate for each TOE component for each SFR. The algorithms supported for keyed-hash message authentication are HMAC-SHA-1 (block size – 512 bits), HMAC-SHA-256 (block size – 512 bits), HMAC-SHA-384 (block size – 1024 bits) and HMAC-SHA-512 (block size – 1024 bits) with key sizes 160, 256, 384 and 512 bits and message digest sizes of 160, 256, 384 and 512 bits respectively. The TOE supports RSA, FFC, and ECDSA in the evaluated configuration. RSA and ECDSA digital signature are used in TLS connections and SSH connections (RSA only). The TOE can be configured to use RSA and ECDSA to authenticate IPsec connections.	

<table>
<tr><th>TOE SFRs</th><th>How the SFR is Satisfied</th></tr>
<tr><td></td><td>
<p>Key generation for asymmetric keys on all models of the TOE implements ECDSA with NIST curve sizes P-256, P-384, and P-521 according to FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.4 and RSA with key sizes 2048 and 3072 bits according to FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.3. Asymmetric cryptographic keys used for IKE peer authentication are generated according to FIPS PUB 186-4, Appendix B.3 for RSA schemes and Appendix B.4 for ECDSA schemes.</p>
<p>Key establishment for asymmetric keys on the TOE implements RSA-based (RSAES-PKCS1-v1_5 as specified in Section 7.2 of RFC 3447), ECDSA-based and DH-based key establishment schemes as specified in NIST SP 800-56A “Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography”. In addition, the TOE also supports DH group 14 key establishment scheme that meets standard RFC 3526, section 3 for interoperability. The TOE's software implementation uses the prime number and generator value specified in RFC 3526 Section 3 when generating parameters for the DH Group 14 key exchange.</p>
<table>
<tr><th>Scheme</th><th>SFR</th><th>Services</th></tr>
<tr><td>RSA</td><td>FCS_TLSS_EXT.1,
FCS_IPSEC_EXT.1,
FCS_SSHS_EXT.1,
FCS_TLSC_EXT.1,
FCS_TLSC_EXT.2</td><td>HTTPS Remote Administration, SSH Remote Administration, syslog over IPsec, Distributed TOE Communication, Syslog over TLS.</td></tr>
<tr><td>ECC (P-256, P-384, P-521)</td><td>FCS_TLSC_EXT.1,
FCS_TLSC_EXT.2,
FCS_IPSEC_EXT.1</td><td>Syslog over TLS, Syslog over IPsec</td></tr>
<tr><td>ECC (P-256, P-384, P-521)</td><td>FCS_TLSS_EXT.1</td><td>HTTPS Remote Administration</td></tr>
<tr><td>FFC</td><td>FCS_TLSC_EXT.1</td><td>Distributed TOE Communication</td></tr>
<tr><td>FFC</td><td>FCS_TLSS_EXT.1</td><td>HTTPS Remote Administration</td></tr>
<tr><td>Diffie-Hellman (Group 14)</td><td>FCS_SSHS_EXT.1
FCS_IPSEC_EXT.1</td><td>SSH Remote Administration, IKE communication.</td></tr>
</table>
<p>The TOE uses a platform-based random bit generator that complies with ISO/IEC 18031:2011 using CTR_DRBG (AES-256) Deterministic Random Bit Generation (DRBG) operating in FIPS mode. In addition, the DRBG is seeded by an entropy source that is at least 256-bit value derived from various highly sensitive and proprietary noise sources described in the proprietary Entropy Design document.</p>
<p>Additionally, the TOE is designed to zeroize secret and private keys when they are no longer required by the TOE. The table in section 7.2 identifies the applicable secret and private keys and summarizes, how they are deleted. The secret keys used for symmetric encryption, private keys, and CSPs used to generate keys, are zeroized immediately after use, or on system shutdown (for all other functions). For plaintext keys: the TOE destroys the reference to the keys stored in volatile memory directly followed by a request for garbage collection; the TOE destroys the abstraction that represents the key for keys stored in non-volatile storage the TSF.</p>
</td></tr>
</table>

TOE SFRs	**How the SFR is Satisfied**
FCS_HTTPS_EXT.1 FCS_TLSC_EXT.1 FCS_TLSC_EXT.2 FCS_TLSS_EXT.1	The TOE implements HTTP over TLS (or HTTPS) to support remote administration on FMC, TLS clients to support secure syslog connections, and TLS server and clients to support FPT_ITT.1. A remote administrator can connect over HTTPS to the TOE with their web browser. FTD supports two different TLS clients that send syslog messages to the external syslog server- FTD TLS client and FTD OS TLS Client. When CC mode is enabled, the TOE is restricted to only support TLSv1.1 and TLSv1.2 for HTTPS sessions and client/server communications between TOE components, with AES 128- or 256-bit symmetric ciphers in CBC and GCM modes, in conjunction with SHA, RSA, and ECDSA. The FMC HTTPS/TLS interface only supports TLSv1.2. The following TLS cipher suites are implemented by the TOE in CC mode: • Relevant to FTP_ITC and FCS_TLSC_EXT.1, **FTD TLS client,** that is configured by the FMC and is the main audit system for audits generated by FTD. It sends audit events such as IPsec and login messages to the external syslog server and Mutual authentication is not supported. Listed in Section 5.3.3.12. • Relevant to FTP_ITC and FCS_TLSC_EXT.2, **FTD OS TLS client**, that is configured through the FTD's command line and sends audit events to an external syslog server such as SSH login, console login, etc. and Mutual authentication is supported. Listed in Section 5.3.3.12. • Relevant to FTP_ITC and FCS_TLSC_EXT.2, for syslog over TLS from FMC/FMCv (client only) as listed in Section 5.3.3.12 of this document. • Relevant to FPT_ITT, FCS_TLSC_EXT.1, and FCS_TLSS_EXT.1 (client and server between FMC and FTD) are as listed in sections 5.3.3.12 and 5.3.3.14 of this document. • Relevant to FTP_TRP.1/Admin and FCS_TLSS_EXT.1 (server only) are as listed in section 5.3.3.14 of this document (FMC only) While the cryptographic modules of the TOE support additional cipher suites (for example, RSA_3DES_EDE_CBC_SHA, RSA_DES_CBC_SHA, RSA_RC4_128_MD5, RSA_RC4_128_SHA, etc.), they are all disabled while operating in CC mode. If the TLS client does not support TLSv1.1 or TLSv1.2, the TLS connection will fail and the administrators will not establish a HTTPS web-based session with the TOE. The Key establishment parameters for each of the TLS connections in the TOE are as follows – 1. FMC/FMCv (HTTPS/TLS)- 2048-bit RSA and ECDHE secp256r1 2. FMC/FMCv and FTD (ITT) – 2048-bit RSA When in CC mode and the TOE acts as a TLS client (e.g., connection to the syslog server), the TOE will verify the server Subject Alternative Name (SAN) against the reference identity (wildcard is supported as required in section 6 of RFC 6125 and per RFC 5280 Appendix A. RFC 5280 is supported for the TLS connection between the distributed TOE components (FMC and FTD) and the attribute type "id-at-title" is used by the TOE client to match the presented identifier with the configured identifier). If verification fails, the TLS connection will not be established. The following NIST curves are presented with the Client Hello by default – secp256r1, secp384r1 and secp521r1. Mutual authentication must be configured with the client-side X.509v3 certificate with RSA 2048-bits (or higher) and SHA-256 (or higher). The key agreement parameters of the server key exchange message are specified in the RFC 5246 (section 7.4.3) for TLSv1.2 and RFC 4346 (section 7.4.3) for TLSv1.1. The TOE conforms to both RFCs. The FMC and FTD must successfully complete a registration process to communicate, which requires administrative actions on the FMC and corresponding administrative actions on the FTD. The administrative actions on FMC and FTD require the administrator to input a

TOE SFRs	How the SFR is Satisfied
	"registration key" that the two devices will use to authenticate their initial TLS communications. During the registration process, the FMC and FTD confirm they have a matching registration key and use their initial self-signed TLS certificates to uniquely identify themselves to each other (each device certificate signed by FMC, including its own, contains a unique identifier stored as an 'id-at-title' attribute, which FMC and FTD each as the unique reference identifier for each other). If the authentication succeeds, the local CA within the FMC will sign and issue a new TLS certificate for the FTD and send (over the existing TLS session) the FTD's new identity certificate and associated keys, and the FMC's root CA cert, and the FMC's root CA certificate and the device certificates which it signed will be used to authenticate all subsequent TLS sessions between the two devices. If device registration fails due to mismatched registration keys, or incorrect IP address or hostname, the information on the FMC and/or FTD needs to be corrected and the registration from FMC reinitiated. TLS session resumption is supported for the following TLS connections of the TOE – the WebUI of the FMC/FMCv. The session tickets used for TLS session resumption are encrypted using symmetric algorithms consistent with FCS_COP.1/DataEncryption claims in this ST – AES used in CBC and GCM modes and key sizes of 128 and 256 bits. The session tickets adhere to the structural format provided in section 4 of RFC 5077.
FCS_IPSEC_EXT.1	**FTD only** The TOE implements IPsec to provide both X509v3 certificate and pre-shared key-based authentications and encryption services to prevent unauthorized viewing or modification of data as it travels over the external network. The TOE implementation of the IPsec standard (in accordance with the RFCs noted in the SFR) uses the Encapsulating Security Payload (ESP) protocol to provide authentication, encryption and anti-replay services. In addition, the TOE supports both transport and tunnel modes. IPsec Internet Key Exchange, also called IKE, is the negotiation protocol that lets two peers agree on how to build an IPsec Security Association (SA). In the evaluated configuration, only IKEv2 is supported. The IKEv2 protocols implement Peer Authentication using the RSA and ECDSA algorithms with X.509v3 certificates or pre-shared keys. IKEv2 separates negotiation into two phases: SA and Child SA. IKE SA creates the first tunnel, which protects later IKE negotiation messages. The key negotiated in IKE SA enables IKE peers to communicate securely in IKE Child SA. During Child SA IKE establishes the IPsec SA. IKE maintains a trusted channel, referred to as a Security Association (SA), between IPsec peers that is also used to manage IPsec connections, including: • The negotiation of mutually acceptable IPsec options between peers (including peer authentication parameters, either signature based or pre-shared key based. • The establishment of additional Security Associations to protect packets flows using Encapsulating Security Payload (ESP), and • The agreement of secure bulk data encryption AES keys for use with ESP. After the two peers agree upon a policy, the security parameters of the policy are identified by an SA established at each peer, and these IKE SAs apply to all subsequent IKE traffic during the negotiation The TOE implements IPsec using the ESP protocol as defined by RFC 4303, using the cryptographic algorithms AES-CBC-128, AES-CBC-256, AES-GCM-128 and AES-GCM-256 (both specified by RFCs 3602 and 4106) along with SHA-based HMAC algorithms, and using IKEv2, as specified for FCS_IPSEC_EXT.1.5, to establish security associations. NAT traversal is supported in IKEv2 by default.

TOE SFRs	How the SFR is Satisfied
	The IKE SA exchanges use only main mode and the IKE SA lifetimes are able to be limited to 24 hours for Phase 1 (SAs) and 8 hours for Phase 2 (Child SAs). Administrators can require use of main mode by configuring the mode for each IPsec tunnel, as in the following examples: Devices > VPN > Site To Site or Devices > VPN > Remote Access IKE Options (click on **IKE** tab) IKEv2 Mode **Tunnel mode** — (default) Encapsulation mode is set to tunnel mode. Tunnel mode applies ESP encryption and authentication to the entire original IP packet (IP header and data), hiding the ultimate source and destination addresses and becoming the payload in a new IP packet. **Transport preferred** — Encapsulation mode is set to transport mode with an option to fallback to tunnel mode if the peer does not support it. In Transport mode only the IP payload is encrypted, and the original IP headers are left intact. **Transport required** — Encapsulation mode is set to transport mode only, falling back to tunnel mode is not allowed. If the endpoints cannot successfully negotiate transport mode, due to one endpoint not supporting it, the VPN connection is not made. **Lifetime (seconds)** – The number of seconds a security association exists before expiring. The default is 28,800 seconds. **Lifetime (kbytes)** – The volume of traffic (in kilobytes) that can pass between IPsec peers using a given security association before it expires. The default is 4,608,000 kilobytes. No specification allows infinite data. In the evaluated configuration, use of "confidentiality only" (i.e. using ESP without authentication) for IPsec connections is prohibited. The TOE allows the administrator to define the IPsec proposal for any IPsec connection to use specific encryption methods and authentication methods as in the following objects: Objects > Object Management > VPN > IKEv2 IPsec Proposal Choose Add IKEv2 IPsec Proposal Enter a **Name** Enter a Description Choose **ESP Hash** method from {sha-1 | sha-256 | sha-384 | sha-512 | null} Choose **ESP Encryption** method from {aes | aes-256 | aes-gcm aes-gcm-256 } **Note:** When AES-GCM is used for encryption, the ESP integrity selection will be "null" because GCM mode provides integrity. AES-GMAC is not allowed in the evaluated configuration. The IKEv2 protocols supported by the TOE implement the following DH groups: 14 (2048-bit MODP), 19 (256-bit Random ECP), 20 (384-bit Random EC), 24 (2048-bit MODP with 256-bit POS) and use the RSA and ECDSA algorithms for Peer Authentication. The following examples are used to specify the DH Group used for SAs: Objects > Object Management > VPN > IKEv2 Policy Choose Add IKEv2 Policy Enter a **Name** Enter a Description Enter a Priority

TOE SFRs	How the SFR is Satisfied
	Enter the **Lifetime** of the SA in seconds. You can specify a value from 120 to 2,147,483,647 seconds. The default is 86400 seconds. Choose **Integrity Algorithms** from **[**sha | sha256 | sha384 | sha512**]** Choose **Encryption Algorithm** from [aes | aes-256 | aes-gcm | aes-gcm-256] Choose **PRF Algorithm** from {sha | sha256 | sha384 | sha512} Add a **DH Group** from {14 | 19 | 20 | 24} The secret 'x' generated is 64 bytes long (or 512 bits), is the same across all the DH groups, and is generated with the DRBG specified in FCS_RBG_EXT.1. This is almost double the size of the highest comparable strength value which is 384 bits. The TOE generates nonces used in IKEv2 exchanges, of at least 128 bits in size and at least half the output size of the negotiated pseudorandom function (PRF) hash. The TOE has a configuration option to deny tunnel if the phase 2 SA is weaker than the phase 1. The crypto strength check is enabled via the **Enable Security Association (SA) Strength Enforcement** checkbox. The TOE can be configured to authenticate IPsec connections using RSA and ECDSA signatures. When using RSA and ECDSA signatures for authentication, the TOE and its peer must be configured to obtain certificates from the same certification authority (CA). Devices > VPN > Site To Site or Devices > VPN > Remote Access IKE Options (click on **IKE** tab) **Policy** - Choose a predefined IKEv2 policy object or create a new one to use. Authentication Type • **Pre-shared Manual Key** — Manually assign the pre-shared key that is used for this VPN. Specify the **Key** and then re-enter it in **Confirm Key** to confirm. When this option is chosen for IKEv2, the **Enforce hex-based pre-shared key only** check box appears, check if desired. If enforced, you must enter a valid hex value for the key, an even number of 2-256 characters, using numerals 0-9, or A-F. • **Certificate** — When you use Certificates as the authentication method for VPN connections, peers obtain digital certificates from a CA server in your PKI infrastructure, and trade them to authenticate each other. To configure an IKEv2 connection to use a RSA or ECDSA signature, select the authenticate type **Certificate**. To define rules for matching the DN or FQDN of the IPsec peer certificate: First, create a certificate map via FMC (Objects > Object Management > VPN > Certificate Map), and add a rule to the certificate map to match the "Alternative Subject" field of the certificate to a value (FQDN/DN). Next, associate the certificate map with the tunnel, depending on tunnel type: • Remote Access VPN (Devices > VPN > Remote Access > Advanced > Certificate Maps > check "Use the configured rules to match a certificate to a Connection Profile > Add Mapping > Certificate Map Name) A crypto map (the Security Policy Definition) set can contain multiple entries, each with a different access list. The crypto map entries are searched in a top-down sequence - the TOE attempts to match the packet to the crypto access control list (ACL) specified in that entry. The crypto ACL can specify a single address or a range of address and the crypto map can be applied to an inbound interface or an outbound interface. When a packet

TOE SFRs	How the SFR is Satisfied
	matches a permit entry in a particular access list, the method of security in the corresponding crypto map of that interface is applied. If the crypto map entry is tagged as ipsecisakmp, IPsec is triggered. The traffic matching the permit crypto ACLs would then flow through the IPSec tunnel and be classified as PROTECTED. Traffic that does not match a permit crypto ACL or match a deny crypto ACL in the crypto map, but is permitted by other ACLs on the interface is allowed to BYPASS the tunnel. Traffic that does not match a permit crypto ACL or match a deny crypto ACL in the crypto map, and is also blocked by other non-crypto ACLs on the interface would be DISCARDED.
FCS_SSHS_EXT.1	The TOE supports SSHv2 with the following encryption algorithms - aes128-cbc, aes256-cbc, AEAD_AES_128_GCM, AEAD_AES_256_GCM, in conjunction with HMAC-SHA1, HMAC-SHA-256, HMAC-SHA-512, AEAD_AES_128_GCM and AEAD_AES_256_GCM for integrity and authenticity, and RSA with diffie-hellman-group14-sha1 for the key exchange method. While DES and 3DES, HMAC-MD5 and HMAC-MD5-96, and diffie-hellman-group-1 and other diffie-hellman-exchange groups are all implemented, they are disabled while the TOE is operating in CC Mode. In addition, SSHv1 is also disabled by default for security reasons. If the SSH client does not support the Approved algorithms or SSH version, the SSH connection will fail and the administrators will not establish an SSHv2 CLI session with the TOE. The TOE supports SSH public-key authentication using ssh-rsa and supports password-based authentication. The TOE ensures and verifies that the SSH client's presented public key matches one that is stored within the TOE’s SSH server's authorized keys file. The TOE uses OpenSSH implementation version 7.6p1 to support the SSHv2 connections. The authentication timeout period is 90 seconds allowing clients to retry only 3 times. In addition, both public-key (RSA) and password-based authentication can be configured with password-based being the default method used. Whenever the timeout period or authentication retry limit is reached, the TOE closes the applicable TCP connection and releases the SSH session resources. As SSH packets are being received, the TOE uses a buffer to build all packet information. Once complete, the packet is checked to ensure it can be appropriately decrypted. However, if it is not complete when the buffer becomes full (256 Kbytes) the packet will be dropped. Note that the TOE manages a tracking mechanism for each SSH session so that it can initiate a new key exchange when either approximately 1 hour of time or 1GB of data is reached. An audit event is generated when a successful SSH rekey occurs when either of the thresholds mentioned occurs. SSH connections will be dropped if the TOE receives a packet larger than 32768 bytes.
FIA_AFL.1	**FMC** FMC provides the administrator the ability to specify the maximum number (can be set differently per account on FMC) of unsuccessful authentication attempts via SSH or WebUI (default is five attempts, configurable from 1-99) before the offending account is locked. The configured limit is the maximum number of allowed consecutive failures, thus the defined number of unsuccessful consecutive authentication attempts that results in locking of accounts is one more than the maximum number of allowed consecutive failures. Only an authorized administrator (with the ‘administrator’ role) can unlock a locked account. By default, the predefined ‘admin’ account is exempt from becoming locked, but that default is overridden when CC mode is enabled. If all admin accounts become locked for any reason, FMC can be accessed locally using password recovery procedures. **FTD** The FTD CLI provides the administrator the ability to specify the maximum number of unsuccessful authentication attempts (configurable from 1-10) before the offending account is locked. Only an authorized administrator (with the ‘administrator’ role) can unlock a

TOE SFRs	How the SFR is Satisfied
	locked account. If all admin accounts become locked for any reason, FTD can be accessed locally using password recovery procedures.
FIA_PMG_EXT.1	The TOE supports the local definition of users with corresponding passwords. The passwords can be composed of any combination of upper and lower-case letters, numbers, and special characters as listed in the SFR. Minimum password length is settable by the Authorized Administrator, and supports passwords of 8 to 32 characters. Password composition rules specifying the types and number of required characters that comprise the password are settable by the Authorized Administrator. Passwords can be configured with a maximum lifetime, configurable by the Authorized Administrator. New passwords can be required to contain a minimum of 4-character changes from the previous password.
FIA_UIA_EXT.1	The TOE requires all users to be successfully identified and authenticated before allowing any TSF mediated actions to be performed. All the TOE components support authentication of Security Administrators according to FIA_UIA_EXT.1 and FIA_UAU_EXT.2. Administrative access to the TOE is facilitated through the TOE's CLI (SSH (password-based and public key-based) or local console in FTD and FMC), or web GUI in FMC. The TOE mediates all administrative actions through the CLI and GUI. The TOE presents a warning banner in accordance with FTA_TAB.1 requirement prior to initiating the identification authentication mechanism for those attempting to access the TOE. Once a potential administrative user attempts to access an administrative interface either locally or remotely, the TOE prompts the user for a user name and password. Only after the administrative user presents the correct authentication credentials will access to the TOE administrative functionality be granted. No access is allowed to the administrative functionality of the TOE until an administrator is successfully identified and authenticated. The TOE provides an automatic lockout when a user attempts to authenticate and enters invalid credentials. After a defined number of authentication attempts fail exceeding the configured allowable attempts, the user is locked out until an authorized administrator can unlock the user account.
FIA_UAU_EXT.2	The TOE provides local password-based authentication mechanisms to FMC and FTD. The process for authentication is the same for administrative access whether administration is occurring via a directly connected console cable or remotely via SSHv2 (password-based or public key-based) or TLS. At initial login the administrative user is prompted to provide a username. After the user provides the username, the user is prompted to provide the administrative password associated with the user account. The TOE then either grants administrative access (if the combination of username and password is correct) or indicates that the login was unsuccessful. The TOE does not provide indication of whether the username or password was the reason for an authentication failure.
FIA_UAU.7	When logging in, the TOE will not echo passwords such that passwords are not inadvertently displayed to the user and any other users that might be able to view the login display. The TOE replaced the entered password character with a "*" character or not show any character at all. This depends on where the user is logging in from, for example, using web GUI versus the SSH client. If the authentication fails, the TOE is designed to not indicate either the username and/or password were incorrect. The error message would just state access denied or unable to authorize access. No other information about the failed login in can be ascertained from the error message. Note also that should a user have their session terminated (e.g., due to inactivity), they are required to successfully re-authenticate, by re-entering their identity and authentication data,

TOE SFRs	How the SFR is Satisfied
	in order to gain access to their session. The authentication data is not cached by the TOE for any reason.
FIA_X509_EXT.1/ITT FIA_X509_EXT.1/Rev FIA_X509_EXT.2(1) FIA_X509_EXT.2(2) FIA_X509_EXT.3	The TOE support X.509v3 certificates as defined by RFC 5280. Public key infrastructure (PKI) credentials, such as private keys and certificates are stored securely. The identification and authentication, and authorization security functions protect an unauthorized user from gaining access to the storage. The validity check for the certificates takes place at session establishment and/or at time of import depending on the certificate type. For example, server certificate is checked at session establishment while CA certificate is checked at both. The TOE conforms to standard RFC 5280 for certificate and path validation (i.e., peer certificate checked for expiration, peer certificate checked if signed by a trusted CA in the trust chain, peer certificate checked for unauthorized modification, peer certificate checked for revocation). The TOE can generate a RSA key pair that can be embedded in a Certificate Signing Request (CSR) created by the TOE. The CSR can be generated at the UI. The TOE can then send the CSR manually to a Certificate Authority (CA) for the CA to sign and issue a certificate. Once the certificate has been issued, the administrator can import the X.509v3 certificate into the TOE. Integrity of the CSR and certificate during transit are assured through the use of digital signature (signing the hash of the TOE's public key contained in the CSR and certificate). CRL is configurable and can be used for certificate revocation check (for FTP_ITC.1 only, thus relevant only to FIA_X509_EXT.1/Rev, not relevant to FIA_X509_EXT.1/ITT as no revocation checking is used for communications between TOE components). Checking is also done for the 'basicConstraints' extension and the 'cA' flag to determine whether they are present and set to TRUE. If they are not, the CA certificate is not accepted as a trust anchor. FMC only supports CRL, while FTD supports use of both CRL and OCSP (including verification of the OCSP signing purpose in the certificate that signs the OCSP response). FTD supports CRL for other purposes, i.e., for validation of syslog server certificates for both the TLS connections to TLS servers. The administrators can configure a trust chain by importing the CA certificate(s) that signed and issued the server (syslog) certificate. This will tell the TOE which CA certificate(s) to use during the validation process. If the TOE does not find the trusted root CA, the TLS connections (FTD TLS client and FTD OS TLS client) to the syslog server will fail. When the TOE cannot establish a connection for the validity check using CRL or the OCSP responder for verification, the FTD OS TLS client and the FMC will accept the certificate when transmitting messages to the syslog server, while all other TOE connections will not accept the certificate and the trusted channel will not be established. When communicating with peers, the TOE uses the default certificate that is configured through the FMC and one that matches the peer's request. For more information, please refer to the CC Supplemental User Guide.
FMT_MOF.1/ ManualUpdate	The TOE restricts the ability to enable, disable, determine and modify the behavior of all of the security functions of the TOE to authorized administrators. The TOE provides the ability for authorized administrators to initiate TOE update, access TOE data, such as audit data, configuration data, security attributes, information flow rules, and session thresholds. **FMC** Only accounts with 'administrator' privilege can upload patches to FMC and initiate installation of patches to FMC or FTD devices (the FMC WebUI is used to manually initiate updates to FMC and FTD). Only accounts with 'administrator' privilege can update system configuration settings related to:

TOE SFRs	How the SFR is Satisfied
	• local logging and remote logging • clock settings • account management including account lockout settings and unlocking accounts (for FMC accounts only) • login banners • cryptographic functionality including SSH (FMC and FTD), TLS (FMC), and IPsec (FTD) • generation of CSRs, and import or delete X.509v3 certificates • IPS functionality **FTD** Only accounts with 'config' privilege can update system configuration settings related to: account management including account lockout settings and unlocking accounts (for FTD accounts only)
FMT_MTD.1/CoreData	The TOE provides a web-based GUI (using HTTPS) management interface and CLI or shell (using SSH or serial connection) (FMC provides the Web GUI and CLI, while the FTD provides a CLI) for all TOE administration, including the policy rule sets, user accounts and roles, and audit functions. The ability to manage various security attributes, system parameters and all TSF data is controlled and limited to those users who have been assigned the appropriate administrative role and privileges associated with those roles. Note that all users created are TOE administrators. **Predefined User Roles** The TOE supports the following predefined user roles: • **Administrators** can set up the appliance's network configuration, manage user accounts, and configure system policies and system settings. The Administrator Role provides access to analysis and reporting features, rule and policy configuration, system management, and all maintenance features. Users with the Administrator role have ALL access rights. Note: For all non-IPS management functions, the only TOE user role is "Administrator". This role is granted when a new user account is created and cannot be changed. The IPS Administrator will also be referred to as the "Administrator". More details on additional IPS roles will be provided in the FMT_SMR.2/IPS section below. The web-based GUI is available on the FMC. The web-based GUI on the FMC is highly recommended for daily management of the FMC and its managed Sensors. Local access to the shell which allows access to the underlying operating system is allowed in the CC evaluated configuration for the initial configuration only. For normal daily operations, the web GUI is still the recommended method.
FMT_MTD.1/CryptoKeys	The TOE only provides the ability for authorized administrators to access TOE data, such as audit data, configuration data, security attributes (such as cryptographic keys and certificates used in IPsec connections), routing tables, and session thresholds.
FMT_SMF.1	The TOE includes the functions necessary to administer the TOE locally and remotely via the administrative interfaces of the FTD (SSH CLI, local console) and FMC (WebUI, SSH CLI, local console). All the management functions that are available to be performed on the TOE local console can also be performed remotely via SSH. No access or service is provided prior to identification and authentication, beyond viewing the login banner.

TOE SFRs	How the SFR is Satisfied
	FMC FMC administrators can perform the following functions: • Login locally via console CLI, and remotely via SSH CLI or TLS WebUI • Configure the access banner (via WebUI) • Configure session inactivity time limits (via WebUI) • Update the FMC and FTD TOE components and verify updates using digital signature prior to installing updates (via WebUI) • Configure authentication failure parameters (via WebUI) • Manage cryptographic keys (via WebUI) • Configure cryptographic functionality (via WebUI) • Configure lifetime for IPsec SAs (via WebUI) • Import X.509v3 certificates (via WebUI) • Configure interaction between TOE components (via WebUI) • Re-enable an administrator account (via WebUI) • Set the time which is used for time-stamps (via WebUI) • Configure the reference identifier for the peer (via WebUI) **FTD** FTD administrators can perform the following functions: • Login locally via console CLI, and remotely via SSH CLI • Configure authentication failure parameters • Import X.509v3 certificates (for syslog servers only) • Configure interaction between TOE components • Re-enable an administrator account • Configure the reference identifier for the peer (for syslog servers only)
FMT_SMR.2	The TOE includes one evaluated role which corresponds to the required 'Security Administrator' described in Section 5.3.5.5.
FPT_SKP_EXT.1	The TOE is designed to not to disclose or store plaintext passwords (e.g., passwords are never recorded in the audit records or display during authentication process). The passwords are stored hashed using Approved SHA-512 with a 32-bit salt value. Only 'root' user account with access to the shell can view the hashed passwords and this is prohibited in the evaluated configuration. Pre-shared keys are stored in plaintext and not visible via any admin interface or in any configuration file even by accounts that have full administrative access such as the default 'admin' account. The same is true for cryptographic keys such as encryption symmetric keys and private keys. The public keys can be viewed but cannot be modified without detection. Note that access to public keys is restricted to administrators.
FPT_APW_EXT.1	The TOE is designed to not to disclose or store plaintext passwords (e.g., passwords are never recorded in the audit records or display during authentication process). The passwords are stored hashed using Approved SHA-512 with a 32-bit salt value. Only 'root' user account with access to the shell can view the hashed passwords and this is prohibited in the evaluated configuration. The same is true for cryptographic keys such as encryption symmetric keys and private keys. The public keys can be viewed but cannot be modified without detection. Note that access to public keys is restricted to administrators.
FPT_STM_EXT.1	The FMC provides a source of date and time information for the TOE, used in audit timestamps, in validating service requests, and for tracking time-based actions related to session management including timeouts for inactive administrative sessions

TOE SFRs	How the SFR is Satisfied
	(FTA_SSL_EXT.*), and renegotiating SAs for IPsec tunnels (FCS_IPSEC_EXT.1). This function can only be accessed from within the configuration exec mode via the privileged mode of operation or using the appropriate role. The clock function is reliant on the system clock provided by the underlying hardware. FMC's clock can be configured manually by the administrators. The FTD must be configured by the administrator to synchronize its clock with the FMC clock.
FPT_TST_EXT.1	The TOE runs a suite of self-tests during initial start-up (power-on-self-tests or POST) to verify its correct operation. When CC mode is enabled on the FMC and FTD, additional cryptographic tests and software integrity test will be run during start-up. The self-testing includes cryptographic algorithm tests (known-answer tests) that feed pre-defined data to cryptographic modules and confirm the resulting output from the modules match expected values, and firmware integrity tests that verify the digital signature of the code image using RSA-2048 with SHA-512. The cryptographic algorithm testing verifies proper operation of encryption functions, decryption functions, signature padding functions, signature hashing functions, and random number generation. The firmware integrity testing verifies the FTD and FMC images have not been tampered with or corrupted. If any of these self-tests fails, the TOE will cease operation. Noise source health tests are run both periodically and at start-up on the FTD to determine the functional health of the noise source. These tests are specifically designed to catch catastrophic losses in the overall entropy associated with the noise source. Tests are run on the raw noise output, before the application of any conditioners. If a noise source fails the health test either at start-up or after the device is operational, the platform will be shut down. Whenever a failure (e.g., POST or integrity test fails) occurs within the FTD that results in the FTD ceasing operation, the FTD securely disables its interfaces to prevent the unintentional flow of any information to or from the FTD and reloads. So long as the failures persist, the FTD will continue to reload. This functionally prevents any failure from causing an unauthorized information flow. There are no failures that circumvent this protection.
FPT_TUD_EXT.1	The TOE components (FMC and FTD) have specific versions that can be queried by an administrator. When updates are made available by Cisco, an administrator can obtain and manually install those updates. Digital signatures (RSA) are used to verify software/firmware update files (to ensure they have not been modified from the originals distributed by Cisco) before they are used to update the applicable TOE components. The update process will fail if the digital signature verification process fails. Updates can be downloaded from https://software.cisco.com with a Cisco.com account. The appropriate software image is then downloaded to the administrator's workstation, then uploaded to FMC (FTD updates are uploaded to FMC then pushed from FMC to FTD). Software update files are verified using digital signatures (RSA) automatically at the time they are uploaded to FMC. Update files will fail to be stored on the device if they fail validation. On FMC, the FMC and FTD updates can uploaded and installed by navigating to System > Updates. Several upload files can remain stored locally on FMC and installed to FMC or FTD later. When updates are initiated, they are applied immediately, and the FMC or FTD will reload automatically with the new software version. That same page also shows the currently running version on FMC. To view the currently running version of any FTD, navigate to Devices > Device Management > then select the device > click on the 'Device' tab.

TOE SFRs	**How the SFR is Satisfied**
FPT_ITT.1, FPT_ITT.1/Join, FPT_ITT.1.1[IPS]	The communication between the FMC and FTD is protected by TLSv1.1 and TLSv1.2. TLS provides authentication, key exchange, encryption and integrity protection of all data transmitted between the TOE components.
FTA_SSL_EXT.1	An administrator can configure maximum inactivity times for both local and remote administrative sessions. When a session is inactive (i.e., no session input) for the configured period of time the TOE will terminate the session, requiring the administrator to log in again to establish a new session when needed. The inactivity times are set at a default of 60 minutes, but an Administrator can configure the inactivity time for the FMC and FTD through the FMC WebUI.
FTA_SSL.3	
FTA_SSL.4	An administrator is able to exit out of both local and remote administrative sessions of the FMC and FTD, effectively terminating the session so it cannot be re-used and will require authentication to establish a new session.
FTA_TAB.1	The TOE provides administrators with the capability to configure advisory banner or warning message(s) that will be displayed prior to completion of the logon process at the local console or via any remote connection (e.g., SSH or HTTPS). The TOE displays an advisory notice and a consent warning message for each administrative method of access: • FMC/FMCv: Console, SSH, and WebUI • FTD: The FTD CLI (SSH) and console
FTP_ITC.1	The TOE uses IPsec and/or TLS to protect communications between itself and remote entities for the following purposes: • The TOE protects transmission of audit records when sending syslog message to a remote audit server by transmitting the messages: o From FMC/FMCv as a TLS client, using X.509v3 certificates for assured identification of the syslog server and with mutual authentication supported. o From FTD as a TLS client (FTD TLS Client), that is configured by the FMC and is the main audit system for audits generated by FTD. It sends audit events such as IPsec and login messages to the external syslog server and mutual authentication is not supported. o From FTD as a TLS client (FTD OS TLS Client), that is configured through the FTD's command line and sends audit events to an external syslog server such as SSH login, console login, etc. and mutual authentication is not supported. o From FTD to an external syslog server over IPsec
FTP_TRP.1/Admin	The TOE uses SSHv2 or HTTPS to provide the trusted path (with protection from disclosure and modification) for all remote administration sessions. Optionally, the FTD supports tunneling the SSH connections in IPsec VPN tunnels (remote VPN client). Remote administration of FMC can be performed using SSH or TLS/HTTPS. Remote administration through the CLI of FTD is via SSH.
Reproduced from the EP_IPS_V2.11	
FAU_GEN.1/IPS[IPS]	For each possible intrusion identified by the system, the TOE will generate an event log, also referred to as an intrusion event and event types are not combined. Each event log will

TOE SFRs	How the SFR is Satisfied
FAU_SAR.1[IPS] FAU_SAR.2[IPS] FAU_SAR.3[IPS] FAU_STG.1[IPS]	include a record of the date, time, type of exploit, and contextual information about the source of the attack and its target. For packet-based events, a copy of the packet or packets that triggered the event is also recorded. Managed Sensors will transmit their events to the FMC where the administrators can view the aggregated data and gain a greater understanding of the attacks against the entire network. The administrators can also deploy the managed Sensors in inline allowing them to configure the Sensors to drop or modify packets that are harmful. The web-based UI is the only way to view the intrusion events (Analysis > Intrusions > Events). The list below describes the intrusion event information that can be viewed, searched, filtered, and sorted by the system. In addition, basic contents such as date, time, and type can also be used to filter and sort. Note only Administrators and Intrusion Admins have access to the intrusion events. **Access Control Policy** The access control policy associated with the intrusion policy where the intrusion, preprocessor, or decoder rule that generated the event is enabled. **Access Control Rule** The access control rule that invoked the intrusion policy that generated the event. Default Action indicates that the intrusion policy where the rule is enabled is not associated with a specific access control rule but, instead, is configured as the default action of the access control policy. This field is blank if intrusion inspection was associated with neither an access control rule nor the default action, for example, if the packet was examined by the default intrusion policy. **Application Protocol** The application protocol, if available, which represents communications between hosts detected in the traffic that triggered the intrusion event. **Application Risk** The risk associated with detected applications in the traffic that triggered the intrusion event: Very High, High, Medium, Low, and Very Low. Each type of application detected in a connection has an associated risk; this field displays the highest risk of those. **Count** The number of events that match the information that appears in each row. Note that the Count field appears only after you apply a constraint that creates two or more identical rows. This field is not searchable. **Destination Continent** The continent of the receiving host involved in the intrusion event. **Destination Country** The country of the receiving host involved in the intrusion event. **Destination IP** The IP address used by the receiving host involved in the intrusion event. **Destination Port / ICMP Code**

TOE SFRs	How the SFR is Satisfied
	The port number for the host receiving the traffic. For ICMP traffic, where there is no port number, this field displays the ICMP code. **Destination User** The User ID for any known user logged in to the destination host. **Device** The managed Sensor where the access control policy was deployed. **Domain** The domain of the Sensor that detected the intrusion. This field is only present if you have ever configured the Firepower Management Center for multitenancy. **Egress Interface** The egress interface of the packet that triggered the event. This interface column is not populated for a passive interface. **Egress Security Zone** The egress security zone of the packet that triggered the event. This security zone field is not populated in a passive deployment. **Generator** The component that generated the event. **Ingress Interface** The ingress interface of the packet that triggered the event. Only this interface column is populated for a passive interface. **Ingress Security Zone** The ingress security zone of the packet that triggered the event. Only this security zone field is populated in a passive deployment. **Inline Result** Actions **Intrusion Policy** The intrusion policy where the intrusion, preprocessor, or decoder rule that generated the event was enabled. **Message** The explanatory text for the event. For rule-based intrusion events, the event message is pulled from the rule. **Priority** The event priority as determined by the Cisco Talos Security Intelligence and Research Group (Talos). The priority corresponds to either the value of the priority keyword or the value for the classtype keyword. For other intrusion events, the priority is determined by the decoder or preprocessor. Valid values are high, medium, and low. **Protocol (search only)** The name or number of the transport protocol used in the connection.

TOE SFRs	How the SFR is Satisfied
	Signature ID The signature used to generate the event. **Snort ID (search only)** Specify the Snort ID (SID) of the rule that generated the event or, optionally, specify the combination Generator ID (GID) and SID of the rule, where the GID and SID are separated with a colon (:) in the format GID:SID. **Source Continent** The continent of the sending host involved in the intrusion event. **Source Country** The country of the sending host involved in the intrusion event. **Source IP** The IP address used by the sending host involved in the intrusion event. **Source Port / ICMP Type** The port number on the sending host. For ICMP traffic, where there is no port number, this field displays the ICMP type. **Source User** The User ID for any known user logged in to the source host. The intrusion events cannot be modified but they can be deleted by the Administrators or Intrusion Admins who have restricted access. When the intrusion events storage is full, the newest data will overwrite the oldest data. There is a feature called Threshold where the administrators can control the number of events that are generated per rule over time. They can limit notification to the specified number of event instances per time period or provide notification once per time period after a specified number of event instances. The administrator must specify if the event instances will be tracked by source or destination IP address, the count or the number of event instances, and the number of seconds for the time period for which event instances are tracked. Note the IPS function cannot be disabled unless the whole system is shutdown. The TOE also will generate all of the required auditable events identified in Table 18 (for FMT_SMF.1/IPS and IPS_NTA_EXT.1 only). All other events in the table are addressed by intrusion events, not auditable events. Please see the CC Supplemental User Guide for more details. The TOE can be configured to generate intrusion events. In addition, all management functions are audited as well. There are certain header fields that should not be used to trigger intrusion events (in Inline mode or Passive mode). Logging events related to these fields would generate a deluge of intrusion audit records that would prevent IPS analysts from figuring out what security incidents occur in their monitored network. In addition, logging these fields will provide no benefits. Per version 2.11 of IPS EP, the following fields can be inspected and if in inline mode, dropped or modified (i.e., normalized): – All checksum fields – TCP Reserved field – TCP Urgent Pointer field

TOE SFRs	How the SFR is Satisfied
	In inline mode, the TOE can count invalid checksum packets that are dropped. The TOE can also count the packets that gets normalized or dropped because of failed normalization.
FMT_SMF.1/IPS[IPS] FMT_MOF.1/IPS[IPS] FMT_MTD.1/IPS[IPS] FMT_SMR.2/IPS[IPS]	The Administrators can deploy intrusion policy with intrusion rules to any interface. An interface, however, can only have one policy applied to that interface. The Administrators can also import vendor-defined signatures from Cisco, create their own intrusion rules, create rules to define which traffic is inspected and analyzed, enable anomaly rules/detections, modify thresholds and threshold duration, and configure white-list/black-list. The IPS Analysts (Intrusion Admins) Administrators can create, modify, or delete intrusion policies but only the IPS Administrators can deploy the policies. Here are the security roles in addition to the all-powerful "Administrator" role. • "IPS Administrator" (or Administrator): Have all privileges and access • "IPS Analyst" (or Intrusion Admin): Have all access to intrusion policies, IPS policies and network analysis privileges but cannot deploy policies • Access Admin: Have all access to access control policies but cannot deploy policies • Discovery Admin: Have all access to network discovery, application detection, and correlation features but cannot deploy policies • Security Analyst: Have all access to security event analysis feature
IPS_ABD_EXT.1[IPS] IPS_IPB_EXT.1[IPS] IPS_NTA_EXT.1[IPS] IPS_SBD_EXT.1[IPS]	**FTD Only** The TOE provides network analysis and intrusion policies as part of the FTD's intrusion detection and prevention system. The term "intrusion detection" generally refers to the process of passively analyzing network traffic for potential intrusions and storing attack data for security analysis. The term "intrusion prevention" includes the concept of intrusion detection but adds the ability to block or alter malicious traffic as it travels across the network. In an intrusion detection/prevention deployment, the TOE examines packets as such: • A network analysis policy governs how traffic is decoded and preprocessed so it can be further evaluated, especially for anomalous traffic that might signal an intrusion attempt. • An intrusion policy uses intrusion and preprocessor rules (sometimes referred to collectively as intrusion rules) to examine the decoded packets for attacks based on patterns or signatures. Without decoding and preprocessing, the TOE could not appropriately evaluate traffic for intrusions because protocol differences would make pattern matching impossible. Network analysis policies govern the traffic-handling tasks: 1. Security Intelligence uses reputation intelligence to quickly block connections to or from IP addresses, URLs, and domain names and is an early phase of access control. 2. Before traffic can be inspected by intrusion policies Security Intelligence *lists* and *feeds* are collections of IP addresses, domain names, and URLs that you can use to quickly filter traffic that matches an entry on a list or feed. • A list is a static collection that can be managed manually. • A feed is a dynamic collection that updates on an interval. Security Intelligence lists/feeds are grouped into: • DNS (Domain names)

TOE SFRs	How the SFR is Satisfied
	• Network (IP addresses) • URLs Predefined global Block lists and Allow lists for domains (DNS), IP addresses (Networks), and URLs are available by default and the Administrators can build on the list. Blacklist and Whitelist options are available on IP address, URL, and DNS requests. Using these rules to block or allow an item adds the item to the appropriate default Global list. By default, Access control and DNS policies use these Global lists. These lists can be applied on a per-policy basis A network analysis policy governs packet processing in phases. First the system decodes packets through the first three TCP/IP layers, then continues with normalizing, preprocessing, and detecting protocol anomalies: • The packet decoder converts packet headers and payloads into a format that can be easily used by the preprocessors and later, intrusion rules. Each layer of the TCP/IP stack is decoded in turn, beginning with the data link layer and continuing through the network and transport layers. The packet decoder also detects various anomalous behaviors in packet headers. • The inline normalization preprocessor reformats (i.e., normalizes) traffic to minimize the chances of attackers evading detection. It prepares packets for examination by other preprocessors and intrusion rules and helps ensure that the packets the system processes are the same as the packets received by the hosts on your network. • Various network and transport layers preprocessors detect attacks that exploit IP fragmentation, perform checksum validation, and perform TCP and UDP session preprocessing. o Various application-layer protocol decoders normalize specific types of packet data into formats that the intrusion rules engine can analyze. Normalizing application-layer protocol encodings allows the system to effectively apply the same content-related intrusion rules to packets whose data is represented differently, and to obtain meaningful results. Conformance to protocols has been verified via compliance testing. o The Modbus and DNP3 SCADA preprocessors detect traffic anomalies and provide data to intrusion rules. The operations associated with the anomaly-based IPS policies are *allow the traffic flow* for any sensor interface in any mode and *allow the traffic flow* and *block/drop the traffic flow* in inline mode. Administrators can define strings to match URLs/URIs, and web page content for pattern-matching. o Several preprocessors allow administrators to detect specific threats, such as IP/TCP/UDP/ICMP port scans, ICMP/TCP flooding, DoS attacks and other rate-based attacks (“frequency”). The administrator can configure threshold[2] that mimics normal expected frequency and configure the TOE to detect and drop events exceeding the configured thresholds. When the system identifies a possible intrusion, it generates an intrusion or preprocessor event (sometimes collectively called intrusion events). Managed Sensors transmit their events to the Firepower Management Center, where the administrators can view the aggregated data and gain a greater understanding of the attacks against their network assets.

[2] Although the term “threshold” is used in the TSS, the TOE’s definition of “threshold” matches the definition of frequency in the ep_ips_v2.11. Therefore, “frequency”, rather than “threshold” has been selected in the IPS_ABD_EXT.1.1 requirement.

TOE SFRs	How the SFR is Satisfied
	In an inline deployment, managed Sensors can also drop packets that are known to be harmful. Each intrusion event in the database includes an event header and contains information about the event name and classification; the source and destination IP addresses; ports; the process that generated the event; and the date and time of the event, as well as contextual information about the source of the attack and its target. For packet-based events, the TOE also logs a copy of the decoded packet header and payload for the packet or packets that triggered the event. The packet decoder, the preprocessors, and the intrusion rules engine can all cause the TOE to generate an event. For examples, • If the packet decoder (configured in the network analysis policy) receives an IP packet that is less than 20 bytes, which is the size of an IP datagram without any options or payload, the decoder interprets this as anomalous traffic. If, later, the accompanying decoder rule in the intrusion policy that examines the packet is enabled, the system generates a preprocessor event. • If the IP defragmentation preprocessor encounters a series of overlapping IP fragments, the preprocessor interprets this as a possible attack and, when the accompanying preprocessor rule is enabled, the system generates a preprocessor event. • Within the intrusion rules engine, most intrusion rules are written so that they generate intrusion events when triggered by packets. Please see section 7.1 for more details on Snort rule. Until the administrator deploy new policies to the network interface, rules in the currently deployed intrusion policies behave as follows: • Disabled rules remain disabled. • Rules set to **Generate Events** continue to generate events when triggered. • Rules set to **Drop and Generate Events** continue to generate events and drop offending packets when triggered. The administrator can set thresholds for individual rules, per intrusion policy, to limit the number of times the system logs and displays an intrusion event based on how many times the event is generated within a specified time period. This can prevent the TOE from being overwhelmed with a large number of identical events. The TOE can also be configured to use intrusion rules to detect various attacks such as Teardrop, Bonk, Ping of Death, etc. The administrators can use pre-defined rule or create custom rule to detect these attacks and many more. Please reference the CC Supplemental User Guide for more details. The administrator can configure the Sensor in either a passive or inline deployment. In a passive (promiscuous) IPS deployment, the Sensor monitors traffic flowing across a network using a switch SPAN or mirror port. The SPAN or mirror port allows for traffic to be copied from other ports on the switch. This provides the system visibility within the network without being in the flow of network traffic. When configured in a passive deployment, the system cannot take certain actions such as blocking or shaping traffic. The administrator can configure one or more physical ports (Gigabit ethernet interfaces) on a managed Sensor as passive interfaces and deploy the intrusion policy to that interface via security zone (i.e., the interface is added to the zone). In an inline IPS deployment, the administrator configures the Sensor transparently on a network segment by binding two ports together. The administrator can configure one or more physical ports (Gigabit ethernet interfaces) on a managed Sensor as inline interfaces then assign a pair of inline interfaces to an inline set. The intrusion policy

TOE SFRs	How the SFR is Satisfied
	is then deployed to that inline set via security zone. The management interface (typically eth0) is separate from the other data monitoring interfaces (used as passive or inline) on the Sensor. It is used to set up and register the Sensor to the FMC.

7 SUPPLEMENTAL TOE SUMMARY SPECIFICATION INFORMATION

7.1 Intrusion Rule Definition

An intrusion rule is a set of keywords and arguments that the system uses to detect attempts to exploit vulnerabilities on your network. As the system analyzes network traffic, it compares packets against the conditions specified in each rule. If the packet data matches all the conditions specified in a rule, the rule triggers. If a rule is an alert rule, it generates an intrusion event. If it is a pass rule, it ignores the traffic. For a drop rule in an inline deployment, the system drops the packet and generates an event. The administrator can view and evaluate intrusion events from the FMC web interface.

All rules contain two logical sections: the rule header and the rule options. The rule header contains:

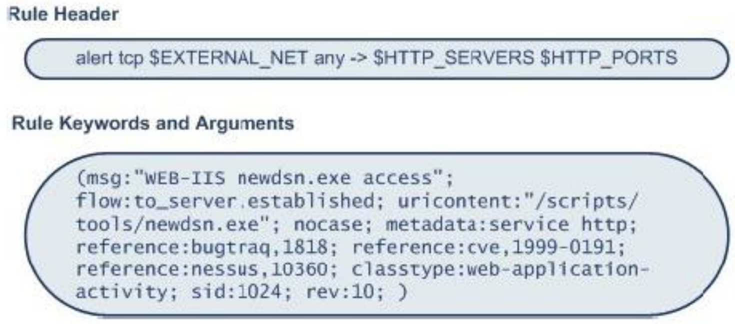
- the rule's action or type
- the protocol
- the source and destination IP addresses and netmasks
- direction indicators showing the flow of traffic from source to destination
- the source and destination ports

The rule options section contains:

- event messages
- keywords and their parameters and arguments
- patterns that a packet’s payload must match to trigger the rule
- specifications of which parts of the packet the rules engine should inspect

The following diagram illustrates the parts of a rule:

For example,



Rule Header

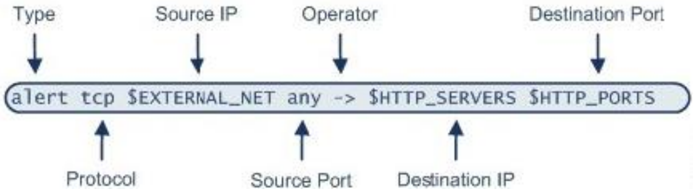
```
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS $HTTP_PORTS
```

Rule Keywords and Arguments

```
(msg:"WEB-IIS newdsn.exe access";
flow:to_server,established; uricontent:"/scripts/
tools/newdsn.exe"; nocase; metadata:service http;
reference:bugtraq,1818; reference:cve,1999-0191;
reference:nessus,10360; classtype:web-application-
activity; sid:1024; rev:10; )
```

7.1.1 Intrusion Rule Header

Every rule has a rule header containing parameters and arguments. The following illustrates parts of a rule header:



```
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS $HTTP_PORTS
```

Action (*alert*) – generates an intrusion event when triggered and operations (allow, block/drop) associated with policies.
Protocol (*tcp*) – Tests TCP traffic only. ICMPv4, ICMPv6, IPv4, IPv6, TCP, and UDP protocols are supported.
Source IP (*$EXTERNAL_NET*) – Tests traffic coming from any host that is not on your internal network.
Source Port (*any*) – Tests traffic coming from any port on the originating host.
Operate (->) – Tests external traffic destined for the web servers on your network.
Destination IP (*$HTTP_SERVERS*) - Tests traffic to be delivered to any host specified as a web server on your internal network. Both IP and IPv6 addresses and ranges are supported.
Destination Port (*$HTTP_PORTS*) - Tests traffic delivered to an HTTP port on your internal network.

7.1.2 Intrusion Rule Options and Keywords

Rule options follow the rule header and are enclosed inside a pair of parentheses. There may be one option or many and the options are separated with a semicolon. If you use multiple options, these options form a logical AND. The action in the rule header is invoked only when all criteria in the options are true. In general, an option may have two parts: a keyword and an argument.

The *message* keyword: Specify meaningful text that appears as a message when the rule triggers.

The *ack* keyword: Specify the acknowledgement value. For example, `(flags: A; ack: 0; msg: "TCP ping detected";)` means receive a TCP packet with the A flag set and the acknowledgement contains a value of 0.

The *content* keyword: Specify data pattern inside a packet. The pattern may be presented in the form of an ASCII string or as binary data in the form of hexadecimal characters.

The *offset* keyword: Specify a certain offset from the start of the data part of the packet to search.

The *dsize* keyword: Specify the length of the data part of a packet.

The *flags* keyword: Find out which flag bits are set inside the TCP header of a packet.

The *fragbits* keyword: Find out which three frag bits (Reserved, Don’t Frag, More Frag) in the IP headers.

The *fragoffset* keyword: Tests the offset of a fragmented packet.

The *itype* keyword: Specify the ICMP type.

The *icode* keyword: Specify the ICMP code.

The *ipopts* keyword: Specify the IP Options. Record Route, Loose Source Routing, Strict Source Routing.

The *ip_proto* keyword: Specify the IP protocol number.

The *id* keyword: Specify the IP header fragment identification field

The *nocase* keyword: Its only purpose is to make a case insensitive search of a pattern within the data part of a packet. It is used in conjunction with the *content* keyword.

The *seq* keyword: Specify the sequence number of a TCP packet.

The *window* keyword: Specify the TCP window size.

The *flow* keyword: Apply a rule on TCP sessions to packets flowing in a particular direction.

The *tos* keyword: Detect a specific value in the Type of Service (TOS) field of the IP header.

The *ttl* keyword: Detect Time to Live value in the IP header of the packet.

7.2 Key Zeroization

The following table describes the key destruction referenced by FCS_CKM.4 provided by the TOE. DRAM (dynamic random access memory) is volatile memory and NVRAM (non-volatile random access memory) is non-volatile "flash" memory.

Table 23: TOE Key Zeroization

Critical Security Parameters (CSPs)	Zeroization Cause and Effect
Diffie-Hellman Shared Secret	Automatically zeroized after completion of DH exchange, by calling a specific API within the two crypto modules, when module is shutdown, or reinitialized. Storage: DRAM Overwritten with: 0x00
Diffie Hellman Private and Public Exponent	Automatically zeroized upon completion of DH exchange, by calling a specific API within the two crypto modules, and when module is shutdown, or reinitialized. Storage: DRAM Overwritten with: 0x00
skeyid	Session Encryption Key and IKE Session Authentication Key. Automatically zeroized after IKE session terminated. Storage: DRAM Overwritten with: 0x00
skeyid_d	Session Encryption Key and IKE Session Authentication Key. Automatically zeroized after IKE session terminated. Storage: DRAM Overwritten with: 0x00
IKE Session Encryption Key	Session Encryption Key and IKE Session Authentication Key. Automatically zeroized after IKE session terminated. Storage: DRAM

Critical Security Parameters (CSPs)	Zeroization Cause and Effect
	Overwritten with: 0x00
IKE Session Authentication Key	Session Encryption Key and IKE Session Authentication Key. Automatically zeroized after IKE session terminated. Storage: DRAM Overwritten with: 0x00
ISAKMP Preshared	Zeroized using the following command: **# no crypto isakmp key** Storage: NVRAM Overwritten with: 0x00
IKE RSA and ECDSA Private and Public Keys	Automatically overwritten when a new key is generated or zeroized using the following commands: **# crypto key zeroize rsa** **# crypto key zeroize ec** Storage: NVRAM Overwritten with: 0x00
IPsec Encryption Key	Automatically zeroized when IPsec session terminated. Storage: DRAM Overwritten with: 0x00
IPsec Authentication Key	Automatically zeroized when IPsec session terminated. Storage: DRAM Overwritten with: 0x00
SSHv2 Private and Public Key	Automatically zeroized upon generation of a new key Storage: NVRAM Overwritten with: 0x00
SSHv2 Session Key	Automatically zeroized when the SSH session is terminated. Storage: DRAM Overwritten with: 0x00

Critical Security Parameters (CSPs)	Zeroization Cause and Effect
All CSPs	Zeroized on-demand on all file systems via the “erase” command. Storage: NVRAM Overwritten with: 0x00
TLS Server Private Key	Zeroized when the HTTPS server is no longer in use. Storage: NVRAM Overwritten with: 0x00

7.3 CAVP Certificate Equivalence

The TOE models and processors included in the evaluation are shown in the following table. The TOE includes multiple cryptographic modules across the range of TOE components. These modules are commonly referred to as FOM (FIPS Object Models). The CAVP-certified FOM of the TOE are listed in the table below (**Table 24**) along with the CPU for which they were certified, and the TOE component on which they’re used. **Table 25** lists the CAVP certificate numbers for each FOM for each applicable SFR.

Table 24: Processors, FOM and CAVP Cert

CPU Family	CPU Model (Microarchitecture)	FOM	Physical Appliances	CAVP Certificate#
FTD				
Intel Xeon D	Intel Xeon D-1526 (Broadwell)	Cisco Security Crypto F6.2	FP 2110	Table 25 Column - Cisco Security Crypto F6.2/Cisco SSL FOM 6.2 (FTD)
	Intel Xeon D-1528 (Broadwell)		FP 2120	
	Intel Xeon D-1548 (Broadwell)	CiscoSSL FOM 6.2	FP 2130	A397
	Intel Xeon D-1577 (Broadwell)		FP 2140	

CPU Family	CPU Model (Microarchitecture)	FOM	Physical Appliances	CAVP Certificate#
Intel Atom C3000	Intel Atom C3558 (Goldmont)		FP 1010	
	Intel Atom C3858 (Goldmont)		FP 1120	
	Intel Atom C3958 (Goldmont)		FP 1140	
FMC				
Intel Xeon E5-2600 v4	Intel Xeon E5-2620 v4 (Broadwell)	CiscoSSL FOM 6.2	FMC4500	A397
	Intel Xeon E5 2640 v4 (Broadwell)		FMC1000 and FMC2500	
Intel Xeon Scalable	Intel Xeon Silver 4110 (Skylake)		FMC1600 and FMC2600	
	Intel Xeon Silver 4116 (Skylake)		FMC4600	
FMCv				
Intel Xeon Scalable w/ Linux 4 on ESXi 6.5	Intel Xeon Bronze 3104 (Skylake) w/ Linux 4 on ESXi 6.5	CiscoSSL FOM 6.2	UCSB-B200-M5, UCSC-C220-M5 and UCSC-C240-M5	A399
	Intel Xeon Silver 4110 (Skylake) w/ Linux 4 on ESXi 6.5		UCSB-B200-M5, UCSC-C220-M5 and UCSC-C240-M5	

CPU Family	CPU Model (Microarchitecture)	FOM	Physical Appliances	CAVP Certificate#
	Intel Xeon® Gold 6128[3] (Skylake) w/ Linux 4 on ESXi 6.5		UCSB-B200-M5, UCSC-C220-M5 and UCSC-C240-M5	
	Intel Xeon Platinum 8153 (Skylake) w/ Linux 4 on ESXi 6.5		UCSB-B200-M5, UCSC-C220-M5 and UCSC-C240-M5	
Intel Xeon E5-2600 v3 w/ Linux 4 on ESXi 6.5	Intel Xeon E5-2620 v3 (Haswell) w/ Linux 4 on ESXi 6.5	CiscoSSL FOM – Virtual 6.2	UCSB-B200-M4, UCSC-C220-M4S, UCSC-C240-M4L, UCSC-C240-M4SX	A971
Intel Xeon E5-2600 v4 w/ Linux 4 on ESXi 6.5	Intel Xeon E5-2609 v4 (Broadwell) w/ Linux 4 on ESXi 6.5	CiscoSSL FOM 6.2	UCSB-B200-M4, UCSC-C220-M4S, UCSC-C240-M4L, UCSC-C240-M4SX	A391
Intel Xeon D w/ Linux 4 on ESXi 6.5	Intel Xeon D-1528 (Broadwell) w/ Linux 4 on ESXi 6.5	CiscoSSL FOM – Virtual 6.2	UCS-E160S-M3	A971
	Intel Xeon D-1548 (Broadwell) w/ Linux 4 on ESXi 6.5		UCS-E180D-M3	

[3] While tested on the Intel Xeon Gold 6130 (Skylake), Intel Xeon Gold 6128 (Skylake) may also be used as part of the evaluated configuration

Table 25: Algorithm Certificate Numbers

Algorithm	SFR	Cisco Security Crypto F6.2/Cisco SSL FOM 6.2 (FTD)	CiscoSSL FOM 6.2 (FMC)	Cisco SSL FOM 6.2/ CiscoSSL FOM – Virtual 6.2 (FMCv)
AES CBC 128/256 GCM 128/256	FCS_COP.1/DataEncryption	4905, A397	A397	A399, A391, A971
RSA 2048/3072 bits Signature Gen & Verify Key Gen	FCS_COP.1/SigGen FCS_CKM.1	2678, A397	A397	A399, A391, A971
DSA 2048/3072 bits	FCS_CKM.1	1304, A397	A397	A399, A391, A971
ECDSA curves P-256, P-384 and P-521 Key Sizes – 256, 384, and 521 bits Signature Gen & Verify Key Gen and Verify	FCS_COP.1/SigGen FCS_CKM.1	1254, A397	A397	A399, A391, A971
Hashing SHA-1, SHA-256, SHA-384, SHA-512	FCS_COP.1/Hash	4012, A397	A397	A399, A391, A971
Keyed Hash	FCS_COP.1/KeyedHash	3272, A397	A397	A399, A391, A971

HMAC-SHA-1, HMAC-SHA-256 HMAC-SHA-384 HMAC-SHA-512				
DRBG CTR_DRBG(AES)	FCS_RBG_EXT.1	1735, A397	A397	A399, A391, A971
KAS ECC KAS FFC CVL	FCS_CKM.2	1520, A397	A397	A399, A391, A971

8 ANNEX A: REFERENCES

The following documentation was used to prepare this ST:

Table 26: References

Identifier	Description
[CC_PART1]	Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and general model, dated April 2017, Version 3.1 Revision 5, CCMB-2017-04-001
[CC_PART2]	Common Criteria for Information Technology Security Evaluation – Part 2: Security functional components, dated April 2017, Version 3.1 Revision 5, CCMB-2017-04-002
[CC_PART3]	Common Criteria for Information Technology Security Evaluation – Part 3: Security assurance components, dated April 2017, Version 3.1 Revision 5, CCMB-2017-04-003
[CEM]	Common Methodology for Information Technology Security Evaluation – Evaluation Methodology, dated April 2017, Version 3.1 Revision 5, CCMB-2017-04-004
[800-38A]	NIST Special Publication 800-38A Recommendation for Block 2001 Edition Recommendation for Block Cipher Modes of Operation Methods and Techniques December 2001
[800-56A]	NIST Special Publication 800-56A, March, 2007 Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography (Revised)
[800-56B]	NIST Special Publication 800-56B Recommendation for Pair-Wise, August 2009 Key Establishment Schemes Using Integer Factorization Cryptography
[FIPS 140-2]	FIPS PUB 140-2 Federal Information Processing Standards Publication Security Requirements for Cryptographic Modules May 25, 2001
[FIPS PUB 186-4]	FIPS PUB 186-3 Federal Information Processing Standards Publication Digital Signature Standard (DSS) July 2013
[FIPS PUB 198-1]	Federal Information Processing Standards Publication The Keyed-Hash Message Authentication Code (HMAC) July 2008
[800-90]	NIST Special Publication 800-90A Recommendation for Random Number Generation Using Deterministic Random Bit Generators January 2012
[FIPS PUB 180-4]	FIPS PUB 180-4 Federal Information Processing Standards Publication Secure Hash Standard (SHS) March 2012

9 ANNEX B: SFR TOE COMPONENTS MAPPING

The following mapping was provided to show which SFR are supported by which TOE component:

Table 27: SFR Mapping

Requirement	Description	Distributed TOE SFR Allocation	Distributed TOE Audit Generation
Reproduced from NDcPP			
FAU_GEN.1	Audit Data Generation	All	All (startup/shutdown, and admin actions)
FAU_GEN.2	User Identity Association	All	N/A
FAU_GEN_EXT.1	Security Audit Generation	All	All
FAU_STG_EXT.1	Protected Audit Event Storage	All	N/A
FAU_STG_EXT.4	Protected Local Audit Event Storage for Distributed TOEs	All	N/A
FAU_STG_EXT.5	Protected Remote Audit Event Storage for Distributed TOEs	All	N/A
FCO_CPC_EXT.1	Communication Partner Control	All	All
FCS_CKM.1	Cryptographic Key Generation	All	N/A
FCS_CKM.2	Cryptographic Key Establishment	All	N/A
FCS_CKM.4	Cryptographic Key Destruction	All	N/A
FCS_COP.1/DataEncryption	Cryptographic Operation (AES Data Encryption/Decryption)	All	N/A
FCS_COP.1/SigGen	Cryptographic Operation (Signature Verification)	All	N/A
FCS_COP.1/Hash	Cryptographic Operation (Hash Algorithm)	All	N/A
FCS_COP.1/KeyedHash	Cryptographic Operation (Keyed Hash Algorithm)	All	N/A
FCS_HTTPS_EXT.1	Protocol Feature Dependent	FMC	FMC
FCS_IPSEC_EXT.1	IPsec Protocol	FTD	FTD

FCS_RBG_EXT.1	Random Bit Generation	All	N/A
FCS_SSHS_EXT.1	SSH Server Protocol	All	All
FCS_TLSC_EXT.1	TLS Client	All	All
FCS_TLSC_EXT.2	TLS Client with authentication	FMC	N/A
FCS_TLSS_EXT.1	TLS Server	All	All
FIA_AFL.1	Authentication Failure Management	All	All
FIA_PMG_EXT.1	Password Management	All	N/A
FIA_UIA_EXT.1	User Identification and Authentication	All	All
FIA_UAU_EXT.2	Password-based Authentication Mechanism	All	All
FIA_UAU.7	Protected Authentication Feedback	All	N/A
FIA_X509_EXT.1/ITT FIA_X509_EXT.1/Rev	X.509 Certification Validation	All	All

FIA_X509_EXT.2(1)
FIA_X509_EXT.2(2)
	X.509 Certificate Authentication	All	N/A

FIA_X509_EXT.3	Certificate Requests	All	N/A
FMT_MOF.1/ManualUpdate	Trusted Update - Management of Security Functions behaviour	All	All
FMT_MTD.1/CoreData	Management of TSF Data	All	N/A
FMT_MTD.1/CryptoKeys	Management of TSF Data	All	N/A
FMT_SMF.1	Specification of Management Functions	FMC (*full*) FTD (*subset*) *(See TSS for details.)*	All
FMT_SMR.2	Restrictions on Security Roles	All	N/A
FPT_SKP_EXT.1	Protection of TSF Data (for reading of all symmetric keys	All	N/A
FPT_APW_EXT.1	Protection of Administrator Passwords	All	N/A
FPT_TST_EXT.1	Testing (Extended)	All	N/A
FPT_ITT.1 FPT_ITT.1/Join	Basic internal TSF data transfer protection	All	All
FPT_STM_EXT.1	Reliable Time Stamps	All	All
FPT_TUD_EXT.1	Trusted Update	All	All
FTA_SSL_EXT.1	TSF-Initiated Session Locking	All	All

FTA_SSL.3	TSF-initiated Termination	All	All
FTA_SSL.4	User-Initiated Termination	All	All
FTA_TAB.1	Default TOE Access Banner	All	N/A
FTP_ITC.1	Inter-TSF Trusted Channel	All	All
FTP_TRP.1/Admin	Trusted Path	All	All
Reproduced from the EP_IPS_V2.11			
FAU_GEN.1/IPS[IPS]	Audit Data Generation (IPS)	All	All
FAU_SAR.1[IPS]	Audit Review (IPS Data)	FMC	N/A
FAU_SAR.2[IPS]	Restricted Audit Review (IPS Data)	FMC	N/A
FAU_SAR.3[IPS]	Selectable Audit Review (IPS Data)	FMC	N/A
FAU_STG.1[IPS]	Protected Audit Trail Storage (IPS Data)	FMC	N/A
FMT_SMF.1/IPS[IPS]	Specification of Management Functions (IPS)	All	FMC
FMT_MOF.1/IPS[IPS]	Management of Security Functions Behavior	All	N/A
FMT_MTD.1/IPS[IPS]	Management of IPS Data	All	N/A
FMT_SMR.2/IPS[IPS]	Security Roles (IPS)	All	N/A
IPS_ABD_EXT.1[IPS]	Anomaly-Based IPS Functionality	FTD	FTD
IPS_IPB_EXT.1[IPS]	IP Blocking	FTD	FTD
IPS_NTA_EXT.1[IPS]	Network Traffic Analysis	All	FMC
IPS SBD_EXT.1[IPS]	Signature-Based IPS Functionality	FTD	FTD
FPT_ITT.1[IPS]	Basic Internal TSF Data Transfer Protection	All	N/A