Cisco Email Security Appliance

Common Criteria Security Target

Version 1.3

29 July 2021

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Table of Contents

Table of Contents 2
List of Tables 5
List of Figures 5
1 SECURITY TARGET INTRODUCTION 9
1.1 ST and TOE Reference 9
1.2 TOE Overview 9
1.2.1 TOE Product Type 10
1.2.2 Supported non-TOE Hardware/ Software/ Firmware 10
1.3 TOE DESCRIPTION 10
1.4 TOE Evaluated Configuration 12
1.5 Physical Scope of the TOE 12
1.6 Logical Scope of the TOE 17
1.6.1 Security Audit 17
1.6.2 Cryptographic Support 18
1.6.3 Identification and authentication 20
1.6.4 Security Management 20
1.6.5 Protection of the TSF 21
1.6.6 TOE Access 21
1.6.7 Trusted path/Channels 22
1.7 Excluded Functionality 22
2 Conformance Claims 23
2.1 Common Criteria Conformance Claim 23
2.2 Protection Profile Conformance 23
2.2.1 Protection Profile Additions 28
2.3 Protection Profile Conformance Claim Rationale 28
2.3.1 TOE Appropriateness 28
2.3.2 TOE Security Problem Definition Consistency 28
2.3.3 Statement of Security Requirements Consistency 29
3 SECURITY PROBLEM DEFINITION 30

3.1 Assumptions 30
3.2 Threats 31
3.3 Organizational Security Policies 32
4 SECURITY OBJECTIVES 33
4.1 Security Objectives for the TOE 33
4.2 Security Objectives for the Environment 33
5 SECURITY REQUIREMENTS 35
5.1 Conventions 35
5.2 TOE Security Functional Requirements 35
5.2.1 Security audit (FAU) 37
5.2.2 Cryptographic Support (FCS) 39
5.2.3 Identification and authentication (FIA) 42
5.2.4 Security management (FMT) 44
5.2.5 Protection of the TSF (FPT) 45
5.2.6 TOE Access (FTA) 46
5.2.7 Trusted Path/Channels (FTP) 47
5.3 TOE SFR Dependencies Rationale for SFRs Found in NDcPPv2.1 48
5.4 Security Assurance Requirements 48
5.4.1 SAR Requirements 48
5.4.2 Security Assurance Requirements Rationale 48
5.5 Assurance Measures 48
6 TOE Summary Specification 50
6.1 TOE Security Functional Requirement Measures 50
7 Annex A: Key Zeroization 67
7.1 Key Zeroization 67
8 Annex B: References 68
9 Annex C: Extended Components Definitions 69
9.1 Security Audit (FAU) 70
9.1.1 Protected audit event storage (FAU_STG_EXT) 70
9.2 Cryptographic Support (FCS) 72
9.2.1 Random Bit Generation (FCS_RBG_EXT) 72
9.2.2 Cryptographic Protocols (FCS_DTLSC_EXT, FCS_DTLSS_EXT, FCS_HTTPS_EXT, FCS_IPSEC_EXT, FCS_NTP_EXT, FCS_SSHC_EXT, FCS_SSHS_EXT, FCS_TLSC_EXT, FCS_TLSS_EXT) 74

9.3 Identification and Authentication (FIA) 82
9.3.1 Password Management (FIA_PMG_EXT) 82
9.3.2 User Identification and Authentication (FIA_UIA_EXT) 83
9.3.3 User authentication (FIA_UAU_EXT) 85
9.3.4 Authentication using X.509 certificates (FIA_X509_EXT) 86
9.4 Protection of the TSF (FPT) 89
9.4.1 Protection of TSF Data (FPT_SKP_EXT) 89
9.4.2 Protection of Administrator Passwords (FPT_APW_EXT.1) 90
9.4.3 TSF Self-Test (FPT_TST_EXT) 91
9.4.4 Trusted Update (FPT_TUD_EXT) 93
9.4.5 Time stamps (FPT_STM_EXT) 97
9.5 TOE Access (FTA) 98
9.5.1 TSF-initiated Session Locking (FTA_SSL_EXT) 98

List of Tables

TABLE 1 ACRONYMS........ 6
TABLE 2 TERMINOLOGY........ 7
TABLE 3 ST AND TOE IDENTIFICATION........ 9
TABLE 4 IT ENVIRONMENT COMPONENTS........10
TABLE 5 HARDWARE MODELS AND SPECIFICATIONS........13
TABLE 6 PROCESSORS AND FOM........18
TABLE 7 FIPS REFERENCES........19
TABLE 8 EXCLUDED FUNCTIONALITY........22
TABLE 9 NIAP TECHNICAL DECISIONS (TD)........23
TABLE 10 PROTECTION PROFILES........28
TABLE 11 TOE ASSUMPTIONS........30
TABLE 12 THREATS........31
TABLE 13 ORGANIZATIONAL SECURITY POLICIES........32
TABLE 14 SECURITY OBJECTIVES FOR THE ENVIRONMENT........33
TABLE 15 SECURITY FUNCTIONAL REQUIREMENTS........36
TABLE 16 AUDITABLE EVENTS........37
TABLE 17 ASSURANCE MEASURES........48
TABLE 18 ASSURANCE MEASURES........48
TABLE 19 HOW TOE SFRS MEASURES........50
TABLE 20 TOE KEY ZEROIZATION........67
TABLE 21 REFERENCES........68
TABLE 22 EXTENDED COMPONENTS........69

List of Figures

FIGURE 1 TOE EXAMPLE DEPLOYMENT........11

Acronyms

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

Table 1 Acronyms

Acronyms / Abbreviations	Definition
AAA	Administration, Authorization, and Accounting
AES	Advanced Encryption Standard
CC	Common Criteria for Information Technology Security Evaluation
CEM	Common Evaluation Methodology for Information Technology Security
CM	Configuration Management
ESA	Email Security Appliance
GCM	Galois Counter Mode
HTTPS	Hyper-Text Transport Protocol Secure
IP	Internet Protocol
IT	Information Technology
NDcPP	collaborative Network Device Protection Profile
PP	Protection Profile
RNG	Random Number Generator
RSA	Rivest, Shamir and Adleman (algorithm for public-key cryptography)
SHS	Secure Hash Standard
SSHv2	Secure Shell (version 2)
ST	Security Target
TCP	Transport Control Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
TLS	Transport Layer Security
TOE	Target of Evaluation
TSC	TSF Scope of Control
TSF	TOE Security Function
TSP	TOE Security Policy

Terminology

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

Table 2 Terminology

Term	Definition
Authorized Administrator	Any user that has been assigned to a privilege level that is permitted to perform all TSF-related functions.
Security Administrator	Synonymous with Authorized Administrator for the purposes of this evaluation.
User	Any entity (human user or external IT entity) outside the TOE that interacts with the TOE.
Firmware (per NIST for FIPS validated cryptographic modules)	The programs and data components of a cryptographic module that are stored in hardware (e.g., ROM, PROM, EPROM, EEPROM or FLASH) within the cryptographic boundary and cannot be dynamically written or modified during execution.

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 Email Security Appliance (ESA) running ESA AsyncOS 13.0. 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, 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]

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 3 ST and TOE Identification

Name	Description
ST Title	Cisco Email Security Appliance Common Criteria Security Target
ST Version	1.3
Publication Date	29 July 2021
Vendor and ST Author	Cisco Systems, Inc.
TOE Reference	Email Security Appliance
TOE Hardware Models	C190, C195, C390, C395, C690, C690X, C695, C695F and the virtual appliances running on UCS platforms, C100v, C300v and C600v on UCS-C220-M4 and UCS-C220-M5
TOE Software Version	ESA AsyncOS 13.0.3-21
Keywords	Email, Data Protection, Authentication, Network Device

1.2 TOE Overview

The TOE, Cisco Email Security Appliance, is a network device. ESA is an appliance that provides comprehensive email protection services for a company’s email system. It is an email protection product that monitors Simple Mail Transfer Protocol (SMTP) network traffic, analyzes the monitored network traffic using various techniques, and reacts to identified threats associated with email messages (such as spam and inappropriate or malicious content). The TOE includes the hardware models as defined in Table 3 in section 1.1.

1.2.1 TOE Product Type

Cisco ESA is a network device that provides connectivity and security services including the capability to secure and control traffic in one device. ESA is designed to serve as a secure SMTP gateway providing the Message Transfer Agent (MTA) role in the customer's network infrastructure. Even though the email protection services are contained within the TOE, this functionality was not evaluated.

ESA provides two management interfaces: A Command Line Interface (CLI) and a web-based Graphical User Interface (GUI). The GUI contains most of the functionality to configure and monitor the TOE. However, not all CLI commands are available in the GUI; some features are only available through the CLI.

1.2.2 Supported non-TOE Hardware/ Software/ Firmware

The TOE supports the following hardware, software, and firmware components in its operational environment. Each component is identified as being required or not based on the claims made in this Security Target. All the following environment components are supported by all TOE evaluated configurations.

Table 4 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 a SSH client installed that is used by the TOE administrator to support TOE administration using the CLI interface through SSH protected channels. Any SSH client that supports SSHv2 may be used.
Management Workstation using web browser for HTTPS	Yes	This includes any IT Environment Management workstation with a web browser installed that is used by the TOE administrator to support TOE administration using the web GUI interface through HTTPS/TLS protected channels. Any web browser that supports TLSv1.1 and TLSv1.2 with the supported ciphersuites may be used.
Local Console	Yes	This includes any IT Environment console that is directly connected to the TOE via the Serial Console Port and is used by the TOE administrator to support TOE administration.
SMTP Server	Yes	This includes any SMTP servers that the TOE receives and sends email traffic. This functionality was not evaluated.
Audit (syslog) Server	Yes	This includes any syslog server to which the TOE would transmit audit log messages using SCP over a secure SSHv2 trusted channel.
CA Server	Yes	This includes any IT Environment CA Server to validate X509 certificates
Update Server	Yes	This includes updates for the potentially malicious files of various types to filter traffic for restricted content. This functionality was not evaluated.

1.3 TOE DESCRIPTION

This section provides an overview of ESA Target of Evaluation (TOE). ESA is a security appliance that is installed between an external network and the customer's internal network. Traffic flowing to and from the external network to the internal network is first routed through the ESA.

The Cisco ESA AsyncOS 13.0 is a Cisco-developed highly configurable proprietary operating system that can detect potentially malicious files of various types, filter traffic for restricted content, and email containing spam messages or phishing attempts. However, this TOE only addresses the functions that provide for the security of the TOE itself as described in Section 1.6 Logical Scope of the TOE below.

The TOE is comprised of both software and hardware. The TOE deployment is ESA running ESA AsyncOS 13.0 software installed on one of the platforms, all of which are described below.

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

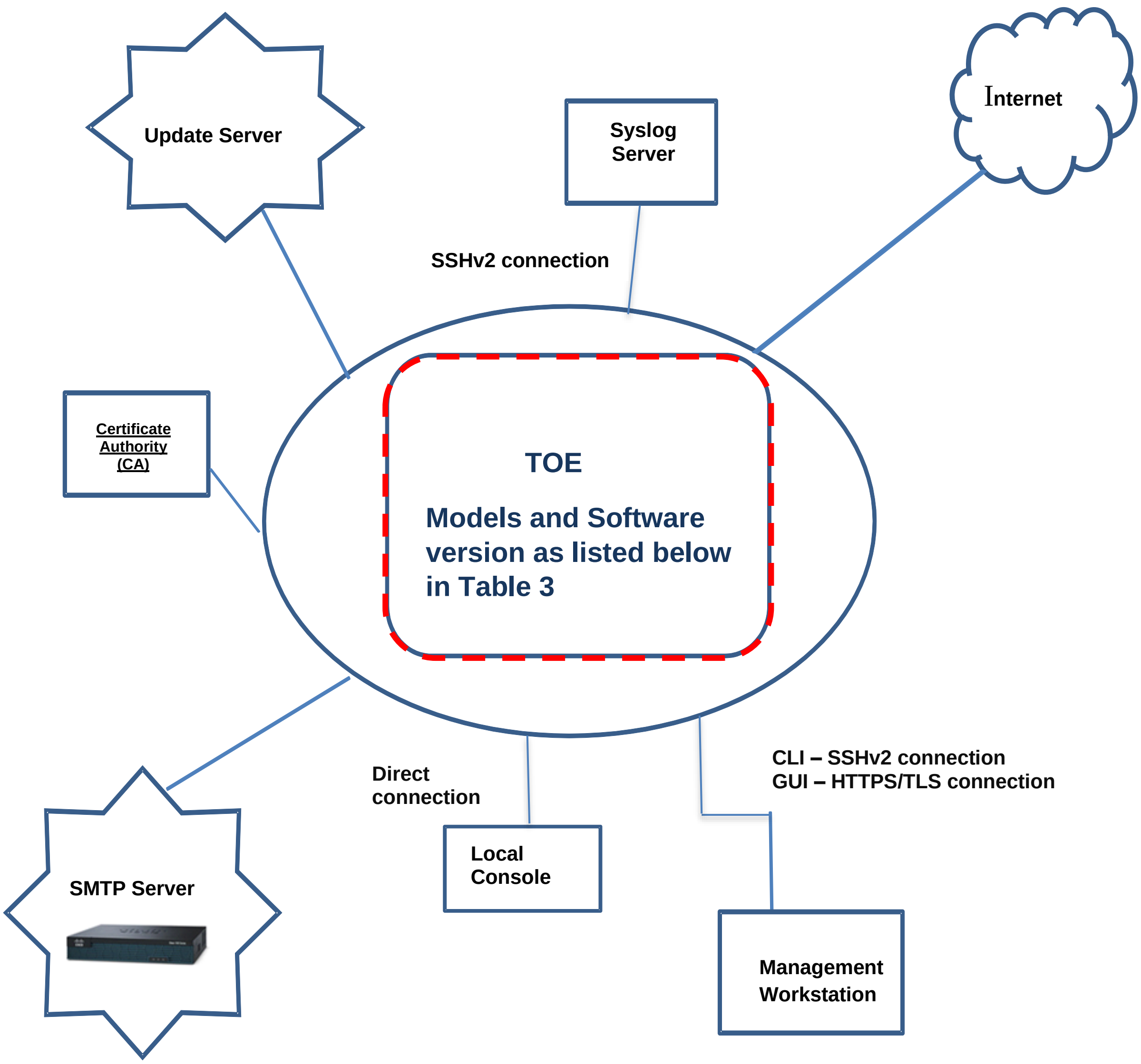


Figure 1 TOE Example Deployment

The previous figure includes the following devices:

- The TOE ESA appliances include:
    - C190, C195, C390, C395, C690, C690X, C695, C695F and the virtual appliances - C100v, C300v and C600v, running on UCS platforms - UCS-C220-M4 and UCS-C220-M5, all running Cisco ESA AsyncOS software version 13.0.3-21
- The following are considered to be in the IT Environment:
    - Local Console to support local Administration (direct connection)
    - Management Workstation to support remote Administration (secure connection is SSHv2 for the CLI and HTTPS/TLS for the GUI)
    - Syslog Server (secure connection is SCP over SSHv2)
    - Update Server
    - Certificate Authority (CA)

1.4 TOE Evaluated Configuration

The TOE consists of one or more appliances as specified in section 1.5 below and includes the ESA AsyncOS software version 13.0.

In addition, if the TOE is to be remotely administered, then the management workstation must be connected to an internal network, SSHv2 must be used to remotely connect to the appliance for the CLI interface and HTTPS/TLS for the GUI interface.

A syslog server is used to store audit records and the connection is secured using SCP over SSHv2. It is recommended that these servers be installed on the internal (trusted) network. The internal (trusted) network is meant to be separated effectively from unauthorized individuals and user traffic, one that is in a controlled environment where implementation of security policies can be enforced.

1.5 Physical Scope of the TOE

The TOE is a hardware and software solution that makes up the Cisco ESA. The TOE hardware is comprised of the following: C190, C195, C390, C395, C690, C690X, C695, C695F and the C100v, C300v, C600v running on Cisco UCS servers. The TOE software is the ESA AsyncOS software version 13.0.

The network, on which they reside, is considered part of the environment.

The TOE is comprised of the following physical specifications as described in Table 5 below.

Table 5 Hardware Models and Specifications

Hardware/Processor/Software	Picture	Size	Power	Interfaces
C190 Intel Xeon E5-2609 v3 (Haswell) processor ESA AsyncOS software version 13.0		1RU: 1.7 x 19 x 31 in. (4.3 x 48.3 x 78.7 cm)	770W Redundant power supply	Two 1-GB Base-T Ethernet LAN ports, can be used as management ports RAID mirroring 10/100/1000 Mbps Two 600-GB hard disk drives (2.5" 10K SAS) hot swappable access for SAS drives One- 8GB DDR4-2133 DIMM1
C195 Intel Xeon Silver 4110 (Skylake) processor ESA AsyncOS software version 13.0		1RU: 2 x 17 x 32 in. (5.1 x 43.2 x 81.3 cm)	770W Redundant power supply	Two 1-GB Base-T Ethernet LAN ports, can be used as management ports RAID mirroring 10/100/1000 Mbps Two 600-GB hard disk drives (2.5" 10K SAS) hot swappable access for SAS drives One- 8GB DDR4-2133 DIMM1
C390 Intel Xeon E5-2620 v3 (Haswell) processor ESA AsyncOS software version 13.0		1RU: 1.7 x 19 x 31 in. (4.3 x 48.3 x 78.7 cm)	770W Redundant power supply	Five 1-Gb Base-T Ethernet LAN ports One management interface (RJ-45), restricted to management use only RAID mirroring 10/100/1000 Two 600 GB hard disk drives (2.5" 10K SAS) hot swappable access for SAS drives Two- 8GB DDR4-2133 DIMM1
C395 Intel Xeon Silver 4116 (Skylake) processor		1RU: 2 x 17 x 32 in. (5.1 x 43.2 x 81.3 cm)	770W Redundant power supply	Six 1-Gb Base-T Ethernet LAN ports One management interface (RJ-45), restricted to management use only RAID mirroring 10/100/1000

Hardware/Processor/ Software	Picture	Size	Power	Interfaces
ESA AsyncOS software version 13.0				Two 600 GB hard disk drives (2.5" 10K SAS) hot swappable access for SAS drives One- 16GB DDR4-2133 DIMM1
C690 Intel Xeon E5-2620 v3 (Haswell) processor ESA AsyncOS software version 13.0		1RU: 2 x 17 x 32 in. (5.1 x 43.2 x 81.3 cm)	770W Redundant power supply	Two 1-GB Base-T Ethernet LAN ports, can be used as management ports RAID mirroring 10/100/1000 Mbps Two 600-GB hard disk drives (2.5" 10K SAS) hot swappable access for SAS drives One- 16GB DDR4-2666 DIMM1
C690X Two Intel Xeon E5-2620 v3 (Haswell) processor ESA AsyncOS software version 13.0		2RU: 3.4 x 19 x 29 in. (8.6 x 48.3 x 73.7 cm)	650W 930W – DC Redundant power supply	Five 1-Gb Base-T Ethernet LAN ports One management interface (RJ-45), restricted to management use only 10/100/1000 Eight 600 GB hard disk (2.5" 10K SAS) hot swappable access for SAS drives Four- 8GB DDR4-2133 DIMM1
C695 Intel Xeon Gold 6126 (Skylake) processor ESA AsyncOS software version 13.0		1RU: 2 x 17 x 32 in. (5.1 x 43.2 x 81.3 cm)	770W Redundant power supply	Six 1-GB Base-T Ethernet LAN ports, can be used as management ports RAID mirroring 10/100/1000 Mbps Eight 600-GB hard disk drives (2.5" 10K SAS) hot swappable access for SAS drives Two- 16GB DDR4-2666 DIMM1
C695F		1RU: 2 x 17 x 32 in. (5.1 x 43.2 x 81.3 cm)	770W Redundant power	Six 1-GB Base-T Ethernet LAN ports, can be used as management ports RAID mirroring

Hardware/Processor/ Software	Picture	Size	Power	Interfaces
Intel Xeon Gold 6126 processor (Skylake) ESA AsyncOS software version 13.0			supply	10/100/1000 Mbps Eight 600-GB hard disk drives (2.5” 10K SAS) hot swappable access for SAS drives Two- 16GB DDR4-2666 DIMM1
C100v, C300v and C600v– installed on UCS C220 M4 Intel® Xeon® E5 2660 v4 Series processor (Broadwell), with VMware ESXi 6.0 Hypervisor, with a single Guest Virtual Machine ESA AsyncOS software version 13.0		1RU: 1.7 x 16.9 x 29.8 in. (4.32 x 43 x 75.6 cm)	Up to two 770 W (AC) hot swappable power supplies or two 1050 W (DC) power supplies. One is mandatory; one more can be added for 1 + 1 redundancy.	Up to 4 LFF or 8 SFF front-accessible, hot-swappable, internal SAS, SATA, or SSD drives, providing redundancy options and ease of serviceability Various PCIe card ports (dependent on which cards are installed), • Virtual Interface Card (VIC) ports, Converged Network Adapter (CNA) ports, Network Interface Card (NIC) ports, Host Bus Adapter (HBA) ports I/O performance and flexibility with one x8 half-height and half-length slot, and one x16 full-height and half-length slot Up to two internal 326GB or two 64GB Cisco FlexFlash drives (SD cards) One internal USB flash drive **Front panel** - One KVM console connector (supplies two USB 2.0 connectors, one GA DB15 connector, and one serial port (RS232) RJ45 connector) **Rear panel** - One DB15 VGA connector, One RJ45 serial port connector, Two USB 3.0 port connectors, One RJ-45 10/100/1000 Ethernet management port, using Cisco Integrated Management Controller (CIMC) firmware, Two Intel i350 embedded (on the motherboard) GbE LOM ports, One flexible modular LAN on motherboard (mLOM) slot that can accommodate various interface

Hardware/Processor/ Software	Picture	Size	Power	Interfaces
				cards
C100v, C300v and C600v– installed on UCS C220 M5 Intel® Xeon® Scalable Platinum 8160M Series processors (Skylake), with VMware ESXi 6.0 Hypervisor, with a single Guest Virtual Machine ESA AsyncOS software version 13.0		**Height** 1.7 in. (4.32 cm) **Width** 16.89 in. (43.0 cm) including handles: 18.98 in. (48.2 cm) **Depth** 29.8 in. (75.6 cm) including handles: 30.98 in. (78.7 cm	Up to two of the following hot-swappable power supplies: 770 W (AC) 1050 W (AC) 1050 W V2 (DC)	**Rear panel** • One 1-Gbps RJ-45 management port (Marvell 88E6176) • Two 10GBase-T LOM ports (Intel X550 controller embedded on the motherboard) • One RS-232 serial port (RJ45 connector) • One DB15 VGA connector • Two USB 3.0 port connectors • One flexible modular LAN on motherboard (mLOM) slot that can accommodate various interface cards **Front panel** • One KVM console connector (supplies two USB 2.0 connectors, one VGA DB15 video connector, and one serial port (RS232) RJ45 connector) **Modular LAN on Motherboard (mLOM) slot** The dedicated mLOM slot on the motherboard can flexibly accommodate the following cards: Cisco Virtual Interface Cards Quad Port Intel i350 1GbE RJ45 Network Interface Card (NIC)

For ordering of the TOE hardware and delivery via commercial carriers, visit Cisco.com Support for the specific model. An example of the ordering details for ESA C690X, see https://www.cisco.com/c/en/us/support/security/email-security-appliance-c690x/model.html

The software is the Cisco AsyncOS software version 13.0.3-21. For ordering and downloading the TOE software, see https://software.cisco.com/#

The TOE guidance documentation that is also considered to be part of the TOE is the Cisco Email Security Appliance (ESA) Common Criteria Configuration Guide document. This document (Cisco Email Security Appliance running AsyncOS 13.0 Common Criteria Operational User Guidance And Preparative Procedures v1.1) is downloadable from the http://cisco.com web site at:https://www.cisco.com/c/en/us/solutions/industries/government/global-government-certifications/common-criteria.html

In Table 1 Common Criteria Certified Product Guidance, enter the certified product name or simply click on the certification date for the product. A PDF version of the document will be displayed, which can be downloaded and saved.

1.6 Logical Scope of the TOE

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

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

These features are described in more detail in the subsections below. In addition, the TOE implements all RFCs of the NDcPP v2.1 as necessary to satisfy testing/assurance measures prescribed therein.

1.6.1 Security Audit

The Cisco Email Security Appliance provides extensive auditing capabilities. The TOE generates a comprehensive set of audit logs that identify specific TOE operations. For each event, the TOE records the date and time of each event, the type of event, the subject identity, and the outcome of the event.

Auditable events include:

- failure on invoking cryptographic functionality such as establishment, termination and failure of cryptographic session establishments and connections;
- modifications to the group of users that are part of the Authorized Administrator roles;
- all use of the user identification mechanism;
- any use of the authentication mechanism;
- Administrator lockout due to excessive authentication failures;
- any change in the configuration of the TOE;
- changes to time;
- initiation of TOE update;
- indication of completion of TSF self-test;
- maximum sessions being exceeded;
- termination of a remote session;
- attempts to unlock a termination session and
- initiation and termination of a trusted channel

The TOE is configured to transmit its audit messages to an SCP server on a remote syslog server. Communication with the syslog server is protected using SCP over SSHv2 and the TOE can determine when communication with the syslog server fails. If the connection fails, the session will need to be reestablished following the configuration settings described in the Cisco Email Security Appliance (ESA) Common Criteria Configuration Guide document.

The audit logs can be viewed on the TOE using the appropriate CLI commands and GUI webpages. The records include the date/time the event occurred, the event/type of event, the user associated with the event, and additional information of the event and its success and/or failure. The TOE does not have an interface to modify audit records, though there is an interface available for the Authorized Administrator to clear audit data stored locally on the TOE.

1.6.2 Cryptographic Support

The TOE provides cryptography in support of other TOE security functionality. All the algorithms claimed have CAVP certificates, based on ESA on the platforms and processors as noted above in Table 5 Hardware Models and Specifications.

The TOE provides cryptography in support of other Cisco ESA security functionality. The ESA software calls the Cisco FIPS Object Module (FOM) v6.2 that has been validated in accordance with the specified standards to meet the requirements listed below and all the algorithms claimed have CAVP certificates.

Refer to Table 7 and Table 7 for algorithm certificate references.

Table 6 Processors and FOM

CPU Family	CPU Model (Microarchitecture)	FOM (CiscoSSL FOM 6.2, CiscoSSL FIPS Object Module 6.2)	Physical Appliances / Platforms	CAVP Certificate#
Intel Xeon E5-2600 v3	Intel Xeon E5-2609 v3 (Haswell)	CiscoSSL FOM 6.2	C190	A405, A406
	Intel Xeon E5-2620 v3		C390, C690, C690X	A405, A406
Intel Xeon Scalable	Intel Xeon Silver 4110 (Skylake)		C195	A397
	Intel Xeon Silver 4116 (Skylake)		C395	A397
	Intel Xeon Gold 6126 (Skylake)		C695, C695F	A402
ESXi 6.0 on Intel® Xeon® Scalable	VMware ESXi 6.0 on Intel® Xeon® Scalable Platinum 8160M (Skylake)	CiscoSSL FIPS Object Module 6.2	C100v, C300v and C600v– installed on UCS C220 M5	C905

CPU Family	CPU Model (Microarchitecture)	FOM (CiscoSSL FOM 6.2, CiscoSSL FIPS Object Module 6.2)	Physical Appliances / Platforms	CAVP Certificate#
ESXi 6.0 on Intel® Xeon® E5 2600 v4	VMware ESXi 6.0 on Intel® Xeon® E5 2660 v4 (Broadwell)		C100v, C300v and C600v– installed on UCS C220 M4	C924

Table 7 FIPS References

Algorithm	Description	Supported Mode	CAVP Cert. #	Module	SFR
AES	Used for symmetric encryption/decryption	CBC (128, 256) CTR (128, 256)	A397, A402, A405, A406, C924, C905	FOM	FCS_COP.1/DataEncryption
SHS (SHA-1, SHA-256, SHA-384 and SHA-512)	Cryptographic hashing services	Byte Oriented	A397, A402, A405, A406, C924, C905	FOM	FCS_COP.1//Hash
HMAC SHA-1	Keyed hashing services and software integrity test	Byte Oriented	A397, A402, A405, A406, C924, C905	FOM	FCS_COP.1/KeyedHash
DRBG	Deterministic random bit generation services in accordance with ISO/IEC 18031:2011	CTR_DRBG (AES 256)	A397, A402, A405, A406, C924, C905	FOM	FCS_RBG_EXT.1
RSA	Signature Verification and key transport	FIPS PUB 186-4 Key Generation, PKCS#1 v.1.5, 2048 bit key,	A397, A402, A405, A406, C924, C905	FOM	FCS_CKM.1 FCS_CKM.2 FCS_COP.1/SigGen

Algorithm	Description	Supported Mode	CAVP Cert. #	Module	SFR
ECDSA	Cryptographic Signature services	FIPS 186-4, Digital Signature Standard (DSS)	A397, A402, A405, A406, C924, C905	FOM	FCS_CKM.1 FCS_CKM.2
CVL – KAS-ECC	Key Agreement	NIST Special Publication 800-56A	A397, A402, A405, A406, C924, C905	FOM	FCS_CKM.2
CVL SSH/TLS	Key Agreement	NIST Special Publication 800-56A	A397, A402, A405, A406, C924, C905	FOM	FCS_CKM.2

The TOE provides cryptography in support of remote administrative management via the SSHv2 for the CLI and HTTPS/TLS for the GUI. SCP over SSHv2 is used to secure the transmission of audit records to the SCP server on the remote syslog server. In addition, the TOE uses the X.509v3 certificate for securing the TLS connections.

The TOE also authenticates software updates to the TOE using a published hash.

1.6.3 Identification and authentication

The TOE provides authentication services for administrative users connecting to the TOE's secure CLI and GUI administrative interfaces using SSHv2 and HTTPS/TLS respectively to secure the connections. Prior to an administrator logging in, a login banner is presented at both the CLI and GUI. The TOE requires Authorized Administrators to be successfully identified and authenticated 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 character complexity rules.

The TOE also provides an automatic lockout when a user attempts to authenticate and enters invalid information. When the threshold for a defined number of authentication attempts fail has exceeded the configured allowable attempts, the user is locked out until an Authorized Administrator can re-enable the user account.

The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for TLS connections.

1.6.4 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 HTTPS/TLS (GUI interface), SSHv2 (CLI interface) session or via a direct local console connection. The TOE provides the ability to securely manage:

- 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 published hash prior to installing those updates;
- Ability to configure the authentication failure parameters;
- Ability to configure the cryptographic functionality;
- Ability to re-enable an Administrator account;
- Ability to configure the audit behavior and
- Ability to set the time

The CLI is the main interface used to administer the TOE since all functionality to configure, securely manage and to monitor the TOE is available via the CLI. The GUI interface can also be used however not all functionality to configure the TOE is available in the GUI. Therefore, in the evaluated configuration it is recommended to use the CLI to perform all configuration and setting of the security functions and to securely mange the TOE.

The TOE supports the security administrator role and is referred to as the Authorized Administrator. Only the Authorized Administrator can perform the above security relevant management functions.

Authorized Administrators can create configurable login banners to be displayed at time of login and can define an inactivity timeout threshold for each admin interface to terminate sessions after a set period of inactivity has been reached.

1.6.5 Protection of the TSF

The TOE protects against interference and tampering by untrusted subjects by implementing identification, authentication, and access controls to limit configuration to Authorized Administrators. The TOE prevents reading of cryptographic keys and passwords. Additionally, Cisco AsyncOS is not a general-purpose operating system and access to Cisco AsyncOS memory space is restricted to only Cisco AsyncOS functions.

The TOE performs testing to verify correct operation of the TOE itself and that of the cryptographic module.

The TOE internally maintains the date and time. This date and time is used as the timestamp that is applied to audit records generated by the TOE. The TOE provides the Authorized Administrators the capability to update the TOE's clock manually to maintain a reliable timestamp.

Finally, the TOE is able to verify any software updates prior to the software updates being installed on the TOE to avoid the installation of unauthorized software.

1.6.6 TOE Access

The TOE can terminate inactive sessions after an Authorized Administrator configurable time-period. Once a session has been terminated, the TOE requires the user to successfully be re-identified and re-authenticate to establish a new session. Sessions can also be terminated if an Authorized Administrator enters the "exit" command.

The TOE can also display an Authorized Administrator specified banner on the CLI and GUI management interfaces prior to allowing any administrative access to the TOE.

1.6.7 Trusted path/Channels

The TOE allows trusted path to be established to itself from remote administrators over SSHv2 for the CLI and HTTPS/TLS for the GUI. The TOE also uses SCP over SSHv2 to push the audit logs to a SCP server on a remote syslog server.

1.7 Excluded Functionality

The following functionality is excluded from the evaluation.

Table 8 Excluded Functionality

Excluded Functionality	Exclusion Rationale
Non-FIPS 140-2 mode of operation	This mode of operation includes non-FIPS allowed operations.
AsyncOS API	Does not include any claimed or in-scope functionality
UCS 240M4, UCS 240M5 and UCS 480M5 platforms for C100v, C300v and C600v	Although UCS 240M4, UCS 240M5 and UCS 480M5 support C100v, C300v and C600v as platforms, they haven't been tested as a part of the evaluation.

This service can be disabled by configuration settings as described in the Cisco Email Security Appliance (ESA) Common Criteria Configuration Guide document. The exclusion of this functionality does not affect the compliance to the collaborative Protection Profile for Network Devices Version 2.1.

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 5.4.

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 Table 10 Protection Profiles below. The following NIAP Technical Decisions (TD) have also been applied to the claims in this document. Each posted TD was reviewed and considered based on the TOE product type, the PP claims and the security functional requirements claimed in this document.

Table 9 NIAP Technical Decisions (TD)

TD Identifier	TD Name	Protection Profiles	References	Publication Date	Applicable?
TD0572	NiT Technical Decision for Restricting FTP_ITC.1 to only IP address identifiers	CPP_ND_V2.1, CPP_ND_V2.2E	FTP_ITC.1	2021.01.29	Yes – TD has been applied.
TD0571	NiT Technical Decision for Guidance on how to handle FIA_AFL.1	CPP_ND_V2.1, CPP_ND_V2.2E	FIA_UAU.1, FIA_PMG_EXT.1	2021.01.29	Yes – TD has been applied.
TD0570	NiT Technical Decision for Clarification about FIA_AFL.1	CPP_ND_V2.1, CPP_ND_V2.2E	FIA_AFL.1	2021.01.29	Yes – TD has been applied.
TD0547	NIT Technical Decision for Clarification on developer disclosure of AVA_VAN	CPP_ND_V2.1, CPP_ND_V2.2E	ND SDv2.1, ND SDv2.2, AVA_VAN.1	2020.10.15	Yes – TD has been applied.
TD0538	NIT Technical Decision for Outdated link to allowed-with list	CPP_ND_V2.1, CPP_ND_V2.2E	Section 2	2020.07.13	Yes - TD has been applied

TD Identifier	TD Name	Protection Profiles	References	Publication Date	Applicable?
TD0536	NIT Technical Decision for Update Verification Inconsistency	CPP_ND_V2.1, CPP_ND_V2.2E	AGD_OPE.1, ND SDv2.1, ND SDv2.2	2020.07.13	Yes - TD has been applied
TD0535	NIT Technical Decision for Clarification about digital signature algorithms for FTP_TUD.1	CPP_ND_V2.1	FTP_TUD.1	2020.07.13	Yes - TD has been applied
TD0533	NIT Technical Decision for FTP_ITC.1 with signed downloads	CPP_ND_V2.1	FTP_ITC.1	2020.07.13	Yes - TD has been applied
TD0532	NIT Technical Decision for Use of seeds with higher entropy	CPP_ND_V2.1	FCS_RGB_EXT.1.2	2020.07.13	Yes - TD has been applied
TD0531	NIT Technical Decision for Challenge-Response for Authentication	CPP_ND_V2.1	FCS_SSHS_EXT.1	2020.07.13	Yes - TD has been applied
TD0530	NIT Technical Decision for FCS_TLSC_EXT.1.1 5e test clarification	CPP_ND_V2.1	FCS_TLSC_EXT.1.1, ND SDv2.1	2020.07.13	No, referenced SFR (FCS_TLSC_EXT.X.1) is not being claimed
TD0529	NIT Technical Decision for OCSP and Authority Information Access extension	CPP_ND_V2.1	FIA_X509_EXT.1/Rev , FIA_X509_EXT.2, ND SD v2.1	2020.07.13	No. OCSP is not claimed in the ST validate the revocation status of the certificate.
TD0528	NIT Technical Decision for Missing EAs for FCS_NTP_EXT.1.4	CPP_ND_V2.1, CPP_ND_V2.2E	FCS_NTP_EXT.1.4, ND SD v2.1, ND SD v2.2	2020.07.13	No, referenced SFR (FCS_NTP_EXT.1) is not being claimed
TD0484	NIT Technical Decision for Interactive sessions in FTA_SSL_EXT.1 & FTA_SSL.3	CPP_FW_V2.0E, CPP_ND_V2.0E, CPP_ND_V2.1	FTA_SSL_EXT.1, FTA_SSL.3	2019.12.18	Yes - TD has been applied

TD Identifier	TD Name	Protection Profiles	References	Publication Date	Applicable?
TD0483	NIT Technical Decision for Applicability of FPT_APW_EXT.1	CPP_FW_V2.0E, CPP_ND_V2.0E, CPP_ND_V2.1	FPT_APW_EXT.1	2019.12.18	Yes - TD has been applied
TD0482	NIT Technical Decision for Identification of usage of cryptographic schemes	CPP_FW_V2.0E, CPP_ND_V2.0E, CPP_ND_V2.1	ND SDv2.0e, FW SDv2.0e, ND SDv2.1, FCS_CKM.2	2019.12.18	Yes - TD has been applied
TD0481	NIT Technical Decision for FCS_(D)TLSC_EXT.X.2 IP addresses in reference identifiers	CPP_FW_V2.0E, CPP_ND_V2.0E, CPP_ND_V2.1	ND SDv2.0, ND SDv2.1, FW SDv2.0e, FAU_GEN.1	2019.12.18	No, referenced SFR (FCS_TLSC_EXT.X.2) is not being claimed
TD0480	NIT Technical Decision for Granularity of audit events	CPP_ND_V2.0E, CPP_ND_V2.1	FIA_AFL.1 (NDcPP), section 2.3.1.3 (ND SD)	2019.12.18	Yes - TD has been applied
TD0478	NIT Technical Decision for Application Notes for FIA_X509_EXT.1 iterations	CPP_FW_v2.0, CPP_FW_V2.0E, CPP_ND_V2.0E, CPP_ND_V2.1	FIA_X509_EXT.1/Rev, FIA_X509_EXT.1/ITT	2019.12.18	Yes - TD has been applied
TD0477	NIT Technical Decision for Clarifying FPT_TUD_EXT.1 Trusted Update	CPP_ND_V2.0E, CPP_ND_V2.1	ND SD V2.0E, ND SD V2.1, FPT_TUD_EXT.1, Tests section	2019.12.18	Yes - TD has been applied
TD0475	NIT Technical Decision for Separate traffic consideration for SSH rekey	CPP_FW_V2.0E, CPP_ND_V2.0E, CPP_ND_V2.1	FCS_SSHC_EXT.1.1, FCS_SSHS_EXT.1.1, ND SD V2.0E, ND SD V2.1	2019.12.18	Yes - TD has been applied
TD0453	NIT Technical Decision for Clarify authentication methods SSH clients can use to authenticate SSH se	CPP_ND_V2.1	FCS_SSHC_EXT.1.9, ND SD v2.1	2019.09.16	Yes - TD has been applied

TD Identifier	TD Name	Protection Profiles	References	Publication Date	Applicable?
TD0451	NIT Technical Decision for ITT Comm UUID Reference Identifier	CPP_FW_V2.0E, CPP_ND_V2.0E, CPP_ND_V2.1	FCS_TLSS_EXT.1.2 and FCS_TLSS_EXT.2.2	2019.09.16	Yes - TD has been applied
TD0450	NIT Technical Decision for RSA-based ciphers and the Server Key Exchange message	CPP_ND_V2.1	FCS_TLSS_EXT.*.3, FCS_DTLSS_EXT.*.4, ND SD v2.1	2019.09.16	Yes - TD has been applied
TD0447	NIT Technical Decision for Using 'diffie-hellman-group-exchange-sha256' in FCS_SSHC/S_EXT.1.7	CPP_FW_V2.0E, CPP_ND_V2.0E, CPP_ND_V2.1	FCS_SSHC_EXT.1.7, FCS_SSHS_EXT.1.7	2019.09.16	Yes - TD has been applied
TD0425	NiT Technical Decision for Cut-and-paste Error for Guidance AA	CPP_ND_V2.0E, CPP_ND_V2.1	ND SD V2.0e, ND SD V2.1, FTA_SSL.3	2019.05.31	Yes - TD has been applied
TD0424	NiT Technical Decision for NDcPP v2.1 Clarification - FCS_SSHC/S_EXT 1.5	CPP_ND_V2.1	ND SD V2.1, FCS_SSHC_EXT.1.5, FCS_SSHS_EXT.1.5	2019.05.31	Yes - TD has been applied
TD0423	NIT Technical Decision for Clarification about application of RfI#201726rev2	CPP_FW_V2.0E, CPP_ND_V2.0E, CPP_ND_V2.1	ND SD V2.0E, FW SD V2.0E, ND SD V2.1	2019.05.31	Yes - TD has been applied
TD0412	NIT Technical Decision for FCS_SSHS_EXT.1.5 SFR and AA discrepancy	CPP_FW_V2.0E, CPP_ND_V2.0, CPP_ND_V2.1	FCS_SSHS_EXT.1.5, ND SD V2.0e, ND SD V2.1	2019.03.22	Yes - TD has been applied
TD0411	NIT Technical Decision for FCS_SSHC_EXT.1.5, Test 1 - Server and client side seem to be confused	CPP_FW_V2.0E, CPP_ND_V2.0E, CPP_ND_V2.1	FCS_SSHC_EXT.1.5, ND SD V2.0E, ND SD V2.1	2019.03.22	Yes - TD has been applied

TD Identifier	TD Name	Protection Profiles	References	Publication Date	Applicable?
TD0410	NIT technical decision for Redundant assurance activities associated with FAU_GEN.1	CPP_ND_V1.0, CPP_ND_V2.0E, CPP_ND_V2.1	FAU_GEN.1, ND SD V1.0, ND SD V2.0e, ND SD V2.1	2019.03.22	Yes - TD has been applied
TD0409	NIT decision for Applicability of FIA_AFL.1 to key-based SSH authentication	CPP_ND_V2.0E, CPP_ND_V2.1	FIA_AFL.1, ND SD v2.0e, ND SD v2.1	2019.03.22	Yes - TD has been applied
TD0408	NIT Technical Decision for local vs. remote administrator accounts	CPP_ND_V2.0E, CPP_ND_V2.1	FIA_UAU_EXT.2, FMT_SMF.1	2019.03.22	Yes - TD has been applied
TD0407	NIT Technical Decision for handling Certification of Cloud Deployments	CPP_FW_V2.0E, CPP_ND_V2.0, CPP_ND_V2.1	FCS_SSHS_EXT.1.5, ND SD V2.0e, ND SD V2.1	2019.03.22	Yes - TD has been applied
TD0402	NIT Technical Decision for RSA-based FCS_CKM.2 Selection	CPP_FW_V2.0E, CPP_ND_V2.0, CPP_ND_V2.1	FCS_CKM.2, ND SD V2.0E, ND SD V2.1	2019.02.24	Yes - TD has been applied
TD0401	NIT Technical Decision for Reliance on external servers to meet SFRs	CPP_ND_V2.0E, CPP_ND_V2.1	FTP_ITC.1	2019.02.24	Yes - TD has been applied
TD0400	NIT Technical Decision for FCS_CKM.2 and elliptic curve-based key establishment	CPP_FW_V2.0E, CPP_ND_V2.0E, CPP_ND_V2.1	FCS_CKM.1, FCS_CKM.2	2019.02.24	Yes - TD has been applied
TD0399	NIT Technical Decision for Manual installation of CRL (FIA_X509_EXT.2)	CPP_ND_V2.0E, CPP_ND_V2.1	FIA_X509_EXT.2, ND SD V2.0E, ND SD V2.1	2019.02.24	Yes - TD has been applied
TD0398	NIT Technical Decision for FCS_SSH*EXT.1.1 RFCs for AES-CTR	CPP_FW_V2.0E, CPP_ND_V2.0E, CPP_ND_V2.1	FCS_SSHC_EXT.1.1, FCS_SSHS_EXT.1.1,	2019.02.24	Yes – TD has been applied

TD Identifier	TD Name	Protection Profiles	References	Publication Date	Applicable?
TD0397	NIT Technical Decision for Fixing AES-CTR Mode Tests	CPP_ND_V2.0E, CPP_ND_V2.1	FCS_COP.1/DataEncryption, ND SD V2.0E, ND SD V2.1	2019.02.24	Yes – TD has been applied
TD0396	NIT Technical Decision for FCS_TLSC_EXT.1.1, Test 2	CPP_ND_V2.0E, CPP_ND_V2.1	FCS_DTLSC_EXT.1.1, FCS_DTLSC_EXT.2.1, FCS_TLSC_EXT.1.1, FCS_TLSC_EXT.2.1, ND SD V2.0E, ND SD V2.1	2019.02.24	No. FCS_TLSC_EXT.1 has not been claimed
TD0395	NIT Technical Decision for Different Handling of TLS1.1 and TLS1.2	CPP_ND_V2.0E, CPP_ND_V2.1	FCS_TLSS_EXT.2.4, FCS_TLSS_EXT.2.5, ND SD V2.0E, ND SD V2.1	2019.02.24	Yes - TD has been applied

Table 10 Protection Profiles

Protection Profile	Version	Date
Network Device Collaborative Protection Profile (NDcPP)	2.1	24 September 2018

2.2.1 Protection Profile Additions

The ST claims exact conformance to the collaborative Protection Profile for Network Devices (NDcPP), Version 2.1. The ST does not include any additions to the functionality described in the NDcPPv2.1.

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, Version 2.1

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 collaborative Protection Profile for Network Devices, Version 2.1 for which conformance is claimed verbatim. All concepts covered in the Protection Profile Security Problem Definition is 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 NDcPP v2.1, for which conformance is claimed verbatim. All concepts covered in the Protection Profile Statement of Security Objectives is 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 the NDcPP v2.1, for which conformance is claimed verbatim. All concepts covered in the Protection Profile Statement of Security Requirements is included in this Security Target. Additionally, the Security Assurance Requirements included in this Security Target are identical to the Security Assurance Requirements included in the NDcPP v2.1.

3 SECURITY PROBLEM DEFINITION

This section 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 11 TOE Assumptions

Assumption	Assumption Definition
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.
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)
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 for particular types of network devices (e.g, firewall).
A.TRUSTED_ADMINISTRATOR	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

Assumption	Assumption Definition
	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.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.

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 12 Threats

Threat	Threat Definition
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_COMMUNICATION_CHANNELS	Threat agents may attempt to target network devices that do not use standardized secure tunneling 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.

Threat	Threat Definition
T.WEAK_AUTHENTICATION_ENDPOINTS	Threat agents may take advantage of secure protocols that use weak methods to authenticate the endpoints – e.g., 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.
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.

3.3 Organizational Security Policies

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

Table 13 Organizational Security Policies

Policy Name	Policy Definition
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.

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.

4.1 Security Objectives for the TOE

The collaborative Protection Profile for Network Devices +Errata 2018314 v2.0e does not define any security objectives for the TOE.

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 14 Security Objectives for the Environment

Environment Security Objective	IT Environment Security Objective Definition
OE.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.
OE.NO_GENERAL_PURPOSE	There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE.
OE.NO_THRU_TRAFFIC_PROTECTION	The TOE does not provide any protection of traffic that traverses it. It is assumed that protection of this traffic will be covered by other security and assurance measures in the operational environment.
OE.TRUSTED_ADMIN	Security Administrators are trusted to follow and apply all guidance documentation in a trusted manner. For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are assumed to monitor the revocation status of all certificates in the TOE's trust store and to remove any certificate from the TOE's trust store in case such certificate can no longer be trusted.
OE.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.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

Environment Security Objective	IT Environment Security Objective Definition
	the equipment is discarded or removed from its operational environment.

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 4, dated: September 2012 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 and claimed PP/EP:

- Unaltered SFRs are stated in the form used in [CC2] or their extended component definition (ECD);
- Refinement made by PP author: Indicated with **bold text** and ~~strikethroughs~~;
- Selection wholly or partially completed in the PP: the selection values (i.e. the selection values adopted in the PP or the remaining selection values available for the ST) are indicated with underlined text
    - e.g. "[selection: *disclosure, modification, loss of use*]" in [CC2] or an ECD might become "disclosure" (completion) or "[selection: disclosure, modification]" (partial completion) in the PP;
- Assignment wholly or partially completed in the PP: indicated with *italicized text*
- Assignment completed within a selection in the PP: the completed assignment text is indicated with *italicized and underlined text*
    - e.g. "[selection: *change_default*, *query*, *modify*, *delete*, [*assignment: other operations*]]" in [CC2] or an ECD might become "change_default, select_tag" (completion of both selection and assignment) or "[selection: change_default, *select_tag*, *select_value*]" (partial completion of selection, and completion of assignment) in the PP;
- Iteration: indicated by adding a string starting with "/" (e.g. "FCS_COP.1/Hash").

Extended SFRs are identified by having a label "EXT" at the end of the SFR name.

Formatting conventions outside of operations and iterations matches the formatting specified within the NDcPPv2.1.

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 15 Security Functional Requirements

Class Name	Component Identification	Component Name
FAU: Security audit	FAU_GEN.1	Audit data generation
	FAU_GEN.2	User Identity Association
	FAU_STG_EXT.1	Protected Audit Event Storage
FCS: Cryptographic support	FCS_CKM.1	Cryptographic Key Generation (for asymmetric keys)
	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
	FCS_RBG_EXT.1	Cryptographic Operation (Random Bit Generation)
	FCS_SSHC_EXT.1	SSH Client Protocol
	FCS_SSHS_EXT.1	SSH Server Protocol
	FCS_TLSS_EXT.1	TLS Server Protocol
FIA: Identification and authentication	FIA_AFL.1	Authentication Failure Management
	FIA_PMG_EXT.1	Password Management
	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/Rev	X.509 Certificate Validation
	FIA_X509_EXT.2	X.509 Certificate Authentication
	FIA_X509_EXT.3	X.509 Certificate Requests
FMT: Security management	FMT_MOF.1/Functions	Management of security functions behaviour
	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_APW_EXT.1	Protection of Administrator Passwords
	FPT_SKP_EXT.1	Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)
	FPT_STM_EXT.1	Reliable Time Stamps
	FPT_TST_EXT.1	TSF Testing
	FPT_TUD_EXT.1	Trusted Update
FTA: TOE Access	FTA_SSL_EXT.1	TSF-initiated Session Locking
	FTA_SSL.3	TSF-initiated Termination
	FTA_SSL.4	User-initiated Termination
	FTA_TAB.1	Default TOE Access Banners
FTP: Trusted path/channels	FTP_ITC.1	Inter-TSF trusted channel
	FTP_TRP.1/Admin	Trusted Path

5.2.1 Security audit (FAU)

5.2.1.1 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 shut-down 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 16*.

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

a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event; and
b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, *information specified in column three of Table 16*.

Table 16 Auditable Events

SFR	Auditable Event	Additional Audit Record Contents
FAU_GEN.1	None.	None.
FAU_GEN.2	None.	None.
FAU_STG_EXT.1	None.	None.
FCS_CKM.1	None.	None.
FCS_CKM.2	None.	None.
FCS_CKM.4	None.	None.
FCS_COP.1/DataEncryption	None.	None.
FCS_COP.1/SigGen	None.	None.
FCS_COP.1/Hash	None.	None.
FCS_COP.1/KeyedHash	None.	None.
FCS_HTTPS_EXT.1	Failure to establish an HTTPS session.	Reason for failure.
FCS_RBG_EXT.1	None.	None.
FCS_SSHC_EXT.1	Failure to establish an SSH session	Reason for failure.
FCS_SSHS_EXT.1	Failure to establish an SSH session	Reason for failure.
FCS_TLSS_EXT.1	Failure to establish an 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.

SFR	Auditable Event	Additional Audit Record Contents
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/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
FIA_X509_EXT.2	None.	None.
FIA_X509_EXT.3	None.	None.
FMT_MOF.1/Functions	None.	None.
FMT_MOF.1/ ManualUpdate	Any attempt to initiate a manual update	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_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).
FPT_TST_EXT.1	None.	None.
FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success and failure)	None.
FTA_SSL_EXT.1	Any attempts at unlocking of an interactive session.	None.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	None.
FTA_SSL.4	The termination of an interactive session.	None.
FTA_TAB.1	None.	None.
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 pathl. Failures of the trusted path functions.	None.

5.2.1.2 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.

5.2.1.3 FAU_STG_EXT.1 Protected Audit Event Storage

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

FAU_STG_EXT.1.2 The TSF shall be able to store generated audit data on the TOE itself [TOE shall consist of a single standalone component that stores audit data locally].

FAU_STG_EXT.1.3 The TSF shall [overwrite previous audit records according to the following rule: [*when allotted space has reached its threshold*]] when the local storage space for audit data is full.

5.2.2 Cryptographic Support (FCS)

5.2.2.1 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 Diffie-Hellman group 14 that meet the following: RFC 3526, Section 3

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

5.2.2.2 FCS_CKM.2 Cryptographic Key Establishment

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

- RSA-based key establishment schemes that meet the following: RSAES-PKCS1-v1_5 as specified in Section 7.2 of RFC 8017, "Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1;
- Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 2, "Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography";
- Key establishment scheme using Diffie-Hellman group 14 that meets the following: RFC 3526, Section 3;

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

5.2.2.3 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, a new value of the key]]*;*
- *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 [single] overwrite consisting of [zeroes]]

that meets the following: *No Standard*.

5.2.2.4 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, CTR] *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, CTR as specified in ISO 10116].

5.2.2.5 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 or greater*],

]
that meet the following: [
- For RSA schemes: FIPS PUB 186-4, "Digital Signature Standard (DSS)", Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSA-PKCS1v1_5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3,

].

5.2.2.6 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*.

5.2.2.7 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] and cryptographic key sizes [160-bit] **and message digest sizes** [160] bits that meet the following: ISO/IEC 9797-2:2011, Section 7-"MAC Algorithm 2".

5.2.2.8 FCS_HTTPS.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 the peer certificate is presented, the TSF shall [not establish the connection] if the peer certificate is deemed invalid.

5.2.2.9 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 [[*1*] software based noise source] with minimum of [256 bits] of entropy at least equal to the greatest security strength, according to ISO/IEC 18031:2011 Table C.1 "Security Strength Table for Hash Functions", of the keys and hashes that it will generate.

5.2.2.10 FCS_SSHC_EXT.1 SSH Client Protocol

FCS_SSHC_EXT.1.1 The TSF shall implement the SSH protocol that complies with RFCs [*4251, 4252, 4253, 4254, 5656, 6668*].

FCS_SSHC_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, [*no other method*]

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

FCS_SSHC_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, aes128-ctr, aes256-ctr*].

FCS_SSHC_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_SSHC_EXT.1.6 The TSF shall ensure that the SSH transport implementation uses [*hmac-sha1*] as its data integrity MAC algorithm(s) and rejects all other MAC algorithm(s).

FCS_SSHC_EXT.1.7 The TSF shall ensure that [*diffie-hellman-group14-sha1, ecdh-sha2-nistp256*] and [*diffie-hellman-group16-sha512, ecdh-sha2-nistp384, ecdh-sha2-nistp521*] are the only allowed key exchange methods used for the SSH protocol.

FCS_SSHC_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.

FCS_SSHC_EXT.1.9 The TSF shall ensure that the SSH client authenticates the identity of the SSH server using a local database associating each host name with its corresponding public key and [*no other methods*] as described in RFC 4251 section 4.1.

5.2.2.11 FCS_SSHS_EXT.1 SSH Server Protocol

FCS_SSHS_EXT.1.1 The TSF shall implement the SSH protocol that complies with RFCs [4251, 4252, 4253, 4254, 5656, 6668].

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

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

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

FCS_SSHS_EXT.1.5 The TSF shall ensure that the SSH transport implementation uses [rsa-sha2-256, rsa-sha2-512] 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] as its data integrity MAC algorithm(s) and rejects all other MAC algorithm(s).

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

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

5.2.2.12 FCS_TLSS_EXT.1 TLS Server Protocol

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

- TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268
- TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246
- TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246
- TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256 as defined in RFC 5246
- TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246

].

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

FCS_TLSS_EXT.1.3 The TSF shall [perform RSA key establishment with key size [2048 bits]; generate Diffie-Hellman parameters of size [2048 bits]].

5.2.3 Identification and authentication (FIA)

5.2.3.1 FIA_AFL.1 Authentication Failure Management

FIA_AFL.1.1 Refinement: The TSF shall detect when an Administrator configurable positive integer within [*1-3*] unsuccessful authentication attempts occur related to *Administrators attempting to authenticate remotely*.

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 authenticating until [*an Authorized Administrator unlocks the locked user account*] is taken by a local Administrator].

5.2.3.2 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: [“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“,”)”, [*no other characters*]];

b) Minimum password length shall be configurable to [*15*] and [*15 or greater*].

5.2.3.3 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.

5.2.3.4 FIA_UAU_EXT.2 Password-based Authentication Mechanism

FIA_UAU_EXT.2.1 The TSF shall provide a local password-based authentication mechanism, and [no other authentication mechanism] to perform local administrative user authentication.

5.2.3.5 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**.

5.2.3.6 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.
- 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 [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.

5.2.3.7 FIA_X509_EXT.2 X.509 Certificate Authentication

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

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

5.2.3.8 FIA_X509_EXT.3 X.509 Certificate Requests

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

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

5.2.4 Security management (FMT)

5.2.4.1 FMT_MOF.1/Functions Management of security functions behaviour

FMT_MOF.1.1/Functions The TSF shall restrict the ability to [determine the behaviour] the functions [transmission of audit data to an external IT entity] to Security Administrators.

5.2.4.1 FMT_MOF.1/ManualUpdate Management of security functions behaviour

FMT_MOF.1/ManualUpdate The TSF shall restrict the ability to enable the functions *to perform manual update to Security Administrators*.

5.2.4.2 FMT_MTD.1/CoreData Management of TSF Data

FMT_MTD.1/CoreData The TSF shall restrict the ability to *manage* the *TSF data to Security Administrators*.

5.2.4.3 FMT_MTD.1/CryptoKeys Management of TSF Data

FMT_MTD.1/CryptoKeys The TSF shall restrict the ability to manage the cryptographic keys to Security Administrators.

5.2.4.4 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* [hash comparison] *prior to installing those updates;*
- *Ability to configure the authentication failure parameters for FIA_AFL.1;*
- [
    - *Ability to configure audit behaviour;*
    - *Ability to manage the cryptographic keys;*
    - *Ability to configure the cryptographic functionality;*
    - *Ability to re-enable an Administrator account;*
    - *Ability to set the time which is used for time-stamps;*
    - *Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors;*
    - *Ability to import X.509v3 certificates to the TOE's trust store;*

  ]

5.2.4.5 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 he TSF shall ensure that the conditions

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

are satisfied.

5.2.5 Protection of the TSF (FPT)

5.2.5.1 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.

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

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

5.2.5.3 FPT_STM.1_EXT.1 Reliable time stamps

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

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

5.2.5.4 FPT_TST_EXT.1: TSF Testing

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

- *RSA Signature Known Answer Test (both signature/verification)*
- *AES Known Answer Test*
- *SHA-1/256/512 Known Answer Test*
- *HMAC Known Answer Test*
- *RNG/DRBG Known Answer Test*
- *Software Integrity Test*

].

5.2.5.5 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 [no other TOE firmware/software version].

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

FPT_TUD_EXT.1.3 The TSF shall provide a means to authenticate firmware/software updates to the TOE using a [published hash] prior to installing those updates.

5.2.6 TOE Access (FTA)

5.2.6.1 FTA_SSL_EXT.1 TSF-initiated Session Locking

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

- terminate the session]

after a Security Administrator-specified time period of inactivity.

5.2.6.2 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*.

5.2.6.3 FTA_SSL.4 User-initiated Termination

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

5.2.6.4 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.

5.2.7 Trusted Path/Channels (FTP)

5.2.7.1 FTP_ITC.1 Inter-TSF trusted channel

FTP_ITC.1.1: The TSF shall **be capable of using [**<u>SSH</u>**] to** provide a trusted communication channel between itself and **authorized IT entities supporting the following capabilities: audit server**, [<u>no other capabilities</u>] 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 **<u>the TSF or the authorized IT entities</u>** to initiate communication via the trusted channel.

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

- *external audit server using SSH*

].

5.2.7.2 FTP_TRP.1 Trusted Path

FTP_TRP.1.1/Admin: The TSF shall **be capable of using [<u>SSH, HTTPS, TLS</u>] to** provide a communication path between itself **and authorized remote administrators** that provides confidentiality and integrity, that is, logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from **<u>disclosure and provides detection of modification of the channel data</u>**.

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

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

5.3 TOE SFR Dependencies Rationale for SFRs Found in NDcPPv2.1

The Security Functional Requirements (SFRs) in this Security Target represent the SFRs identified in the NDcPPv2.1. As such, the NDcPPv2.1 SFR dependency rationale is deemed acceptable since the PP itself has been validated.

5.4 Security Assurance Requirements

5.4.1 SAR Requirements

The TOE assurance requirements for this ST are taken directly from the NDcPPv2.1 which are derived from Common Criteria Version 3.1, Revision 5, dated April 2017. The assurance requirements are summarized in the table below.

Table 17 Assurance Measures

Assurance Class	Components	Components Description
Security Target (ASE)	ASE_CCL.1	Conformance claims
	ASE_ECD.1	Extended components definition
	ASE_INT.1	ST introduction
	ASE_OBJ.1	Security objectives for the operational environment
	ASE_REQ.1	Stated security requirements
	ASE_SPD.1	Security Problem Definition
	ASE_TSS.1	TOE summary specification
Development (ADV)	ADV_FSP.1	Basic Functional Specification
Guidance Documents (AGD)	AGD_OPE.1	Operational user guidance
	AGD_PRE.1	Preparative User guidance
Life Cycle Support (ALC)	ALC_CMC.1	Labeling of the TOE
	ALC_CMS.1	TOE CM coverage
Tests (ATE)	ATE_IND.1	Independent testing - conformance
Vulnerability Assessment (AVA)	AVA_VAN.1	Vulnerability analysis

5.4.2 Security Assurance Requirements Rationale

The Security Assurance Requirements (SARs) in this Security Target represent the SARs identified in the NDcPPv2.1. As such, the NDcPPv2.1 SAR rationale is deemed acceptable since the PP itself has been validated.

5.5 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 18 Assurance Measures

Component	How requirement will be met
Security Target (ASE) / ASE_CCL.1 / ASE_ECD.1 / ASE_INT.1 / ASE_OBJ.1 / ASE_REQ.1 / ASE_SPD.1 / ASE_TSS.1	Section 2 of this ST includes the TOE and ST conformance claim to CC Version 3.1, Revision 5, dated: April 2017, CC Part 2 extended and CC Part 3 conformant and NDcPPv2.1 and the rationale of how TOE provides all of the functionality at a level of security commensurate with that identified in NDcPPv2.1. Section 2 also includes the consistency rationale for the TOE Security Problem Definition and the Security Requirements to include the extended components definition.
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 behaviour 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 ST.
AGD_PRE.1	The Installation Guide describes the installation, generation and startup procedures so that the users of the TOE can setup the components of the TOE in the evaluated configuration.
ALC_CMC.1 ALC_CMS.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.
ATE_IND.1	Cisco will provide the TOE for testing.
AVA_VAN.1	Cisco will provide the TOE for testing.

6 TOE SUMMARY SPECIFICATION

6.1 TOE Security Functional Requirement Measures

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

Table 19 How TOE SFRs Measures

TOE SFRs	How the SFR is Met
FAU_GEN.1	The TOE generates an audit record that is stored internally within the TOE whenever an audited event occurs. Audit records are stored in files within the file system provided by the TOE. The TOE stores auditable events in separate log files containing related types of audited data. The following log files together comprise the TSF audit trail by covering all events listed in Table 16: • Audit Logs - Records AAA (Authentication, Authorization, and Accounting) events. Records all Authorized Administrator interaction with the GUI and command-line interfaces, and captures committed changes of all management activities performed using the GUI interface • Authentication Framework Logs - Records authentication events • CLI Audit Logs - Records a historical audit of command line interface activity related to all management activities performed using the CLI interface • GUI Logs - Records history of the web interface activity related to all management activities • System Logs – Records system errors and all management commit (changes) activity • Updater Logs – Records a history of system and other management updates Note that the TOE generates various other log files that record information about the behavior of the TOE, but these do not contain logs that satisfy the TOE's auditing requirements. The TOE shall ensure that each auditable event is associated with the user that triggered the event and as a result they are traceable to a specific user. For example, a human user, user identity, or related session ID would be included in the audit record. For an IT entity or device, the IP address, MAC address, host name, or other configured identification is presented. Each audit record includes date and time of the audited event, type of event, subject identity, and the outcome (success or failure) of the event. The auditable events comprise: • Start-up and shutdown of the audit function - recorded in System Logs • Access to the TOE and System data - recorded in: CLI Audit Logs (for console interfaces) and GUI logs; and Updater logs (TOE updates). • Reading of information from the audit records - recorded in CLI Audit Logs and HTTP logs for GUI • Unsuccessful attempts to read information from the audit records - recorded in CLI Audit Logs and HTTP logs for GUI • All modifications to the audit configuration that occur while the audit collection functions are operating - recorded in CLI Audit Logs and HTTP logs for GUI • All modifications in the behavior of the functions of the TSF, that include all administrative actions, such as login/logout, generating/import of, changing, or deleting of cryptographic keys (including a reference of any associated keys), resetting of passwords- recorded in CLI Audit Logs and HTTP logs for GUI • All modifications to the values of TSF data, that include all administrative actions, such as login/logout, generating/import of, changing, or deleting of

TOE SFRs	How the SFR is Met
	cryptographic keys (including a reference of any associated keys), resetting of passwords - recorded in CLI Audit Logs and HTTP logs for GUI • Modifications to the group of users that are part of an Administrator role - recorded in CLI Audit Logs and HTTP logs for GUI Authorized Administrators can access all audit information. The Authorized Administrators can manually download the log files by clicking a link to the log directory on the Log Subscriptions page, then clicking the log file to access. Depending on the browser, an Authorized Administrator can view the file in a browser window, or open or save it as a text file. This method uses the HTTP(S) protocol and is the default retrieval method. Example audit events are included below: < Date and time of the event> < type of event> < source IP> <subject identity> <outcome> <url accessed with return HTTP headers> Thu Nov 1 19:03:00 2012 Info: login:10.65.79.90 user:admin session:XtL50wP9GB92YfjVerYb Thu Nov 1 19:03:00 2012 Info: req:10.65.79.90 user:- id:XtL50wP9GB92YfjVerYb 303 POST /login HTTP/1.1 Mozilla/5.0 (Macintosh; Intel Mac OS X 10_7_4) AppleWebKit/537.4 (KHTML, like Gecko) Chrome/22.0.1229.94 Safari/537.4 Thu Nov 1 19:03:02 2012 Info: req:10.65.79.90 user:admin id:XtL50wP9GB92YfjVerYb 200 GET /monitor/user_report HTTP/1.1 Mozilla/5.0 (Macintosh; Intel Mac OS X 10_7_4) AppleWebKit/537.4 (KHTML, like Gecko) Chrome/22.0.1229.94 Safari/537.4 Thu Nov 1 19:03:03 2012 Info: req:10.65.79.90 user:admin id:XtL50wP9GB92YfjVerYb 200 GET /scfw/1y-8.0.0-366/navigation.css HTTP/1.1 Mozilla/5.0 (Macintosh; Intel Mac OS X 10_7_4) AppleWebKit/537.4 (KHTML, like Gecko) Chrome/22.0.1229.94 Safari/537.4 Thu Nov 1 19:03:03 2012 Info: req:10.65.79.90 user:admin id:XtL50wP9GB92YfjVerYb 200 GET /scfw/1y-8.0.0-366/widget/tablecols/table- cols-min.css HTTP/1.1 Mozilla/5.0(Macintosh; Intel Mac OS X 10_7_4) AppleWebKit/537.4 (KHTML, like Gecko) Chrome/22.0.1229.94 Safari/537.4 Thu Nov 1 19:03:03 2012 Info: req:10.65.79.90 user:admin id:XtL50wP9GB92YfjVerYb 200 GET /yui_webui HTTP/1.1 Mozilla/5.0 (Macintosh; Intel Mac OS X 10_7_4) AppleWebKit/537.4 (KHTML, like Gecko) Chrome/22.0.1229.94 Safari/537.4 Thu Nov 1 19:03:04 2012 Info: req:10.65.79.90 user:admin id:XtL50wP9GB92YfjVerYb 200 GET /javascript?CSRFKey=f0fadf9c-fce3-43b6- 84ae-3b42f559bcd5&language=en-usHTTP/1.1 Mozilla/5.0 (Macintosh; Intel Mac OS X 10_7_4) AppleWebKit/537.4 (KHTML, like Gecko) Chrome/22.0.1229.94 Safari/537.4
FAU_GEN.2	The TOE shall ensure that each auditable event is associated with the user that triggered the event and as a result they are traceable to a specific user. For example, a human user, user identity, or related session ID would be included in the audit record. For an IT entity or device, the IP address, MAC address, host name, or other configured identification is presented. A sample audit record is below: Fri Jun 6 16:35:22 2014 Info: login:192.168.1.228 user:admin session:fI023Y0gCdc5u4BuWqL8 The HTTPS session has been established successfully.
FAU_STG_EXT.1	The TOE is configured to send the audit log records within each of the log files listed below to a specified, SCP server on a remote syslog server. The TOE protects

TOE SFRs	How the SFR is Met
	communications with the remote syslog server via SCP over SSHv2. This must be configured by an Authorized Administrator. Once configured, the TOE can automatically send the audit records to the configured SCP server on a remote syslog server. The log files that must be configured to be sent to the external syslog server are: • Audit Logs • Authentication Logs • CLI Audit Logs • GUI Logs • System Logs Note that the TOE can also export various other log file's audit records to an external syslog server, but these other log files do not contain logs that satisfy the TOE's auditing requirements. The TOE provides the following mechanisms for sending the log files to a remote syslog server: • SCP on Remote [syslog] Server - a remote syslog server that supports an scp command can copy log files from the TOE to the remote syslog server. The user of the scp command on the remote syslog server must be the Authorized Administrator on the TOE, as the TOE will prompt for the Authorized Administrator password before processing the SCP request • SCP Push - additionally, the TOE can be configured to periodically push log files to a SCP server on a remote syslog server The Authorized Administrator can configure the time interval for sending the log files to the remote syslog with a minimum time lapse of 60 seconds and maximum time of 12 days. The time setting is customized based on day, hour, minutes and seconds. There is also a configurable maximum log file size limit (100KB – 104MB configuration range) for sending logs. If the log file size crosses the limit before the configured time duration has expired, the logs will still get pushed. Both of the above SCP methods are secured by SCP over SSHv2. The SCP is the method that periodically pushes log files to an SCP server on a remote syslog server. This method requires an SSH SCP server on the remote syslog server using SSHv2 protocol. The subscription requires a username, SSH key, and destination directory on the remote syslog server. Log files are transferred based on a rollover schedule set by the Authorized Administrator. The TOE generates an email alert to the Authorized Administrator and begins overwriting the oldest stored audit records when the audit trail becomes full. (Note that the TOE does not stop collecting or producing System data). The alert is generated to an Authorized Administrator who has been configured via the Command Line Interface (*alertconfig* command) to receive email alerts for this event. The TOE does not provide interfaces to modify individual records. When the audit trail becomes full, the TOE ensures that the most recent audit records will be maintained, limited only by the available storage space. The SCP push method periodically pushes log files to an SCP server on a remote syslog server. This method also requires an SSH SCP server on a remote syslog server using SCP over SSHv2 protocol to secure the connection. The subscription requires a username (recommend that it is Authorized Administrator on the TOE), SSH key and destination directory on the remote syslog server. Log files are transferred based on a rollover schedule set by the Authorized Administrator. The TOE is capable of detecting when the SSH connection fails. If the connection fails, the session will need to be reestablished following the configuration settings described in the Cisco Email Security Appliance (ESA) Common Criteria Configuration Guide document.

TOE SFRs	How the SFR is Met
	The TOE also stores a local set of audit records on the TOE and continues to do so if the communication with the syslog server goes down. Once the connection is restored, the audit records will be sent to the remote syslog server as configured. For example, on the next SCP push based on either the maximum log file size being exceeded or on the time interval, the current log file and the log files previously unsuccessfully transferred will be transferred. The TOE stores the audit logs locally as configured with the *logconfig* command in the CLI and the Log Subscriptions page in the GUI. The size of the local log files is set by an Authorized Administrator using the 'Rollover by File Size' configuration setting. Once the file reaches the specified size, they are sent to the remote syslog server using SCP over SSHv2. These transfers can also be configured based on configured time intervals. Only Authorized Administrators are able to clear the local logs, and there is no TOE interface that allows for administrators to modify the contents of the local audit records. The TOE's default installation configures the audit log files to maintain 10 files of no more than 10MB for each log subscription. The Authorized Administrator does not need to configure this setting however, this value is customizable. The Authorized Administrators can configure each log subscription to allow 1-1000 maximum log files, and each log file can be configurable to a maximum of between 100KB and 100MB. There is no limit to the number of log subscriptions that the Authorized Administrator can create. With a typical configuration, the log space should not grow beyond a reasonable limit. If through customization of the log limits, the log files grow too much, alerts will be sent to the Authorized Administrators when the log partition grows beyond 90% usage. If the space available for storing audit records is exhausted, the TOE will start to overwrite the oldest records in the audit trail and generate an email alert to this effect and send it to an Authorized Administrators. Refer to the Cisco Email Security Appliance Common Criteria Configuration Guide for full details and configuration settings.
FCS_CKM.1 FCS_CKM.2	The TOE implements Diffie-Hellman based key establishment schemes that meets RFC 3526, Section 3. The TOE implements and uses the prime and generator specified in RFC 3526 Section 3 when generating parameters for the key exchange. In addition, ECC schemes are used with P-256, P-384 and P-521. The TOE complies with section 5.6 and all subsections regarding asymmetric key pair generation in the NIST SP 800-56A and with section 6 and all subsections regarding RSA key pair generation. The TOE employs RSA-based key establishment, RSAES-PKCS1-v1_5 used in cryptographic operations as specified in Section 7.2 of RFC 8017. The TOE can create a RSA public-private key pair of 2048 bit or greater that can be used to generate a Certificate Signing Request (CSR). Via offline CSR the TOE can send the CSR to a Certificate Authority (CA) for the CA to generate a certificate; and receive its certificate (including X.509v3) from the CA. The Integrity of the CSR and certificate during transit are assured through use of digital signatures (encrypting the hash of the TOE's public key contained in the CSR and certificate). The TOE can store and distribute the certificate to external entities including Registration Authorities (RA). The TOE can also use X.509v3 certificates for authentication of TLS sessions. The TOE acts as both a sender and receiver for RSA-based key establishment schemes 800-56A and 800-56B. The key pair generation portions of "The RSA Validation System" for FIPS 186-4 were used

<table>
<tr><th>TOE SFRs</th><th>How the SFR is Met</th></tr>
<tr><td></td><td>as a guide in testing the FCS_CKM.1.

TOE acts as both a sender and receiver for Diffie-Helman and RSA based key establishment schemes.

<table>
<tr><th>Scheme</th><th>SFR</th><th>Service</th></tr>
<tr><td>RSA</td><td>FCS_TLSS_EXT.1
FCS_SSHC/S_EXT.1</td><td rowspan="4">Remote Administration</td></tr>
<tr><td>DH (group 14)</td><td>FCS_TLSS_EXT.1
FCS_SSHC/S_EXT.1</td></tr>
<tr><td>RSAES-PKCS1</td><td>FCS_TLSS_EXT.1</td></tr>
<tr><td>ECC</td><td>FCS_SSHC/S_EXT.1</td></tr>
<tr><td>RSA</td><td>FCS_SSHC/S_EXT.1</td><td rowspan="3">Remote Syslog Server</td></tr>
<tr><td>DH (group 14)</td><td>FCS_SSHC/S_EXT.1</td></tr>
<tr><td>ECC</td><td>FCS_SSHC/S_EXT.1</td></tr>
</table>

For details on each protocol see the related SFR.</td></tr>
<tr><td>FCS_CKM.4</td><td>The TOE meets all requirements as specified by the cryptographic key destruction method of the keys and the Critical Security Parameters (CSPs) when no longer required for use.

The TOE zeroizes all the cryptographic keys used within the TOE after the key is no longer of use to the TOE. The cryptographic module performs the overwrite of the cryptographic keys and other critical security parameters that are handled by the CiscoSSL library (FOM) are zeroized using a function that will overwrite the memory once they are no longer in use.

Swap space is encrypted using AES to avoid accidental leakage of CSPs. As part of the reload command, an option to wipe the data is provided. The wipe option along with the 'wipedata' command will overwrite the hard drive with zeros so that the keys are zeroized within the old core dump files.

The information provided in 7.1 Key Zeroization includes all the secrets, keys and associated values, the description, and the method used to zeroization when no longer required for use. This information is provided in the reference section for ease and readability of all the all secrets, keys and associated values, their description and zeroization methods.</td></tr>
<tr><td>FCS_COP.1/DataEncryption</td><td>The TOE provides symmetric encryption and decryption capabilities using AES in CBC and CTR mode (128 and 256 bits) as described in ISO 18033-3 and ISO 10116.

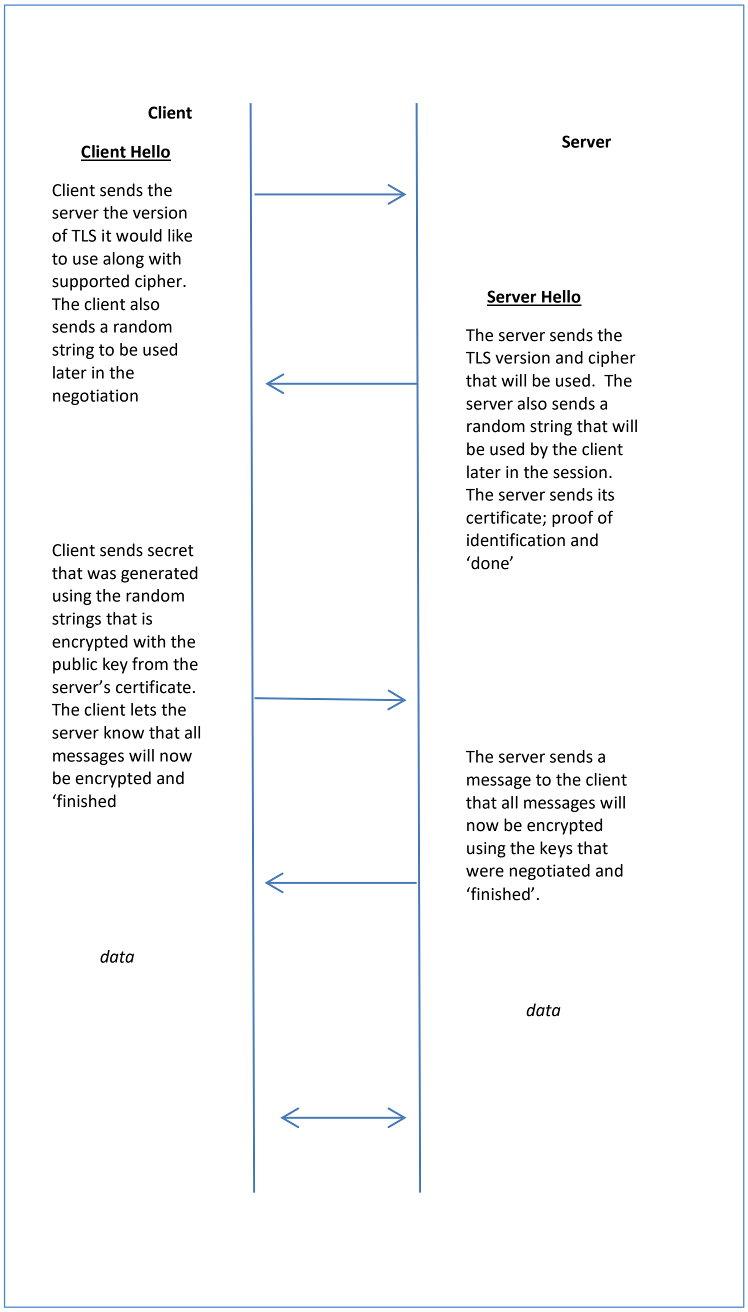
See CAVP certificate in Table 7 FIPS References for validation details.

AES is implemented in the following protocols: TLSv1.1, TLSv1.2 and SSHv2.

The TOE also provides AES encryption and decryption in support of SSHv2 and TLSv1.1/2 for secure communications.</td></tr>
</table>

TOE SFRs	How the SFR is Met
	The configuration and management of the cryptographic algorithms is provided through the CLI, to include the auditing of configuring the options by the Authorized Administrator. The relevant FIPS certificate numbers are listed in Table 7 FIPS References
FCS_COP.1/SigGen	The TOE provides cryptographic signature services using RSA Digital Signature Algorithm with key size of 2048 and greater as specified in FIPS PUB 186-4, "Digital Signature Standard". The relevant FIPS certificate numbers are listed in Table 7 FIPS References. The TOE provides cryptographic signatures in support of SSHv2 and TLSv1.1/2 for secure communications. The TOE provides the RSA option in support of SSHv2 and TLSv1.1/2 key establishment. RSA 2048-bit is used in the establishment of both TLSv1.1/2 and SSHv2 key establishment. For SSHv2, RSA host keys are supported Management of the cryptographic algorithms is provided through the CLI with auditing of those commands. The relevant FIPS certificate numbers are listed in Table 6 Algorithm Certificate References
FCS_COP.1/Hash FCS_COP.1/KeyedHash	The TOE provides cryptographic hashing services using SHA-1, SHA-256, SHA-384, and SHA-512 as specified in ISO/IEC 10118-3:2004. The TOE provides hashing as part of the TLS session integrity. In addition, SHA-384 hashing is used for verification of software image integrity. The TOE uses server-side X.509v3 certificates for authentication. Digital signature is comprised of implementing an encrypted hash function. Verification of the digital signature includes the process of decrypting the encrypted hash and verifying the hash is valid. SHA1 is also used in the keyed hash function of HMAC. The TOE provides Secure Hash Standard (SHS) hashing in support of TLS, for secure communications. Management of the cryptographic algorithms is provided through the CLI with auditing of those commands. The TOE provides keyed-hashing message authentication services using HMAC-SHA-1, key size 160 bits, and message digest sizes 160 bits as specified in ISO/IEC 9797-2:2011, Section 7 "MAC Algorithm 2". The block size for HMAC-SHA1 is 512 bits. The TOE provides SHS hashing and HMAC message authentication in support of SSHv2, and TLSv1.1/2 for secure communications. Management of the cryptographic algorithms is provided through the CLI with auditing of those commands. SHS hashing and HMAC message authentication (SHA-1) is used in the establishment of HTTPS, TLS and SSHv2 sessions. Refer to the Cisco Email Security Appliance Common Criteria Configuration Guide for full details and configuration settings.
FCS_HTTPS_EXT.1	The TOE implements HTTPS over TLS as specified in RFC 2818 and FCS_TLSS_EXT.1. The TSF HTTPS implementation authenticates the TOE to the remote client with an X.509 certificate. Authorized Administrators manage the TOE identity certificates using the Destination Controls page in the GUI or *Interfaceconfig* command in the TOE CLI. HTTPS then uses the Authorized Administrators selected identity certificate. The TSF HTTPS implementation performs server-based authentication using a server X.509v3 certificate to establish the TLS session. The TSF HTTPS implementation does not require client authentication at the TLS level but presents the Web interface logon page for Authorized Administrators to authenticate using their name and password.

TOE SFRs	How the SFR is Met
FCS_RBG_EXT.1	The TOE implements a NIST-approved AES-CTR Deterministic Random Bit Generator (DRBG), as specified in SP 800-90 seeded by an entropy source that accumulates entropy from a TSF-software based noise source as described in FCS_RBG_EXT.1. This output is used directly to seed the DRBG. The deterministic RBG is seeded with a minimum of 256 bits of entropy, which is at least equal to the greatest security strength of the keys and hashes that it will generate.
FCS_SSHC_EXT.1	The TOE implements SSHv2 to secure the remote session between the TOE and syslog server. SSHv2 is implemented according to the following RFCs: 4251, 4252, 4253, 4254, 5656, 6668. The TOE supports public key-based authentication. The TOE uses a SCP push to securely send the audit logs to a remote syslog server over a secured SSHv2 session. The SSH client (the TOE) authenticates the identity of the SSH server (remote syslog server) using a local database associating each host name with its corresponding public key as described in RFC 4251 section 4.1. SSH connections will be dropped if the TOE receives a packet larger than 256KB (262,144 bytes). Large packets are detected by the SSH implementation and dropped internal to the SSH process. A rekey occurs after a threshold of no longer than one hour and no more than one gigabyte of transmitted data. The key exchange methods allowed by the TOE in the evaluated configuration are, diffie-hellman-group14-sha1, ecdh-sha2-nistp256, diffie-hellman-group16-sha512, ecdh-sha2-nistp384 and ecdh-sha2-nistp521. Any session where the SSH server offers only non-compliant algorithms or key sizes will be rejected by the SSH client. SSH sessions can only be established when compliant algorithms and key sizes can be negotiated. Noting the SSH client only negotiates ssh-rsa during hostkey negotiation. The TOE implementation of SSHv2 supports the following: • public key algorithms for authentication: ssh-rsa • encryption algorithms, aes128-cbc, aes256-cbc, aes128-ctr and aes256-ctr to ensure confidentiality of the session. • hashing algorithms HMAC-SHA1 to ensure the integrity of the session.
FCS_SSHS_EXT.1	The TOE implements SSHv2 for remote CLI sessions. SSHv2 is implemented according to the following RFCs: 4251, 4252, 4253, 4254, 5656, 6668. The TOE supports both public key-based and password-based authentication. The sessions keys for SSH sessions have a threshold of one hour, and no more than one gigabyte of transmitted data. A rekey is performed whenever either of the two thresholds is reached. SSH connections will be dropped if the TOE receives a packet larger than 256KB (262,144 bytes). Large packets are detected by the SSH implementation and dropped internal to the SSH process. The key exchange methods used by the TOE is a configurable option, however only diffie-hellman-group14-sha1, ecdh-sha2-nistp256 and ecdh-sha2-nistp521 are the only allowed methods within the evaluated configuration. Any session where the SSH client offers only non-compliant algorithms or key sizes will be rejected by the SSH server. SSH sessions can only be established when compliant algorithms and key sizes can be negotiated. The TOE implementation of SSHv2 supports the following: • public key algorithms for authentication: rsa-sha2-256 and, rsa-sha2-512. • Public key-based and password-based authentication for administrative users

TOE SFRs	How the SFR is Met
	accessing the TOE's CLI through SSHv2. • encryption algorithms, aes128-cbc, aes256-cbc, aes128-ctr and aes256-ctr to ensure confidentiality of the session. • hashing algorithms HMAC-SHA1 is used to ensure the integrity of the session.
FCS_TLSS_EXT.1	An Authorized Administrator can initiate inbound TLSv1.1 and TLSv1.2 connections using the web-based GUI for remote administration of the TOE. Following is the TLS handshake and exchange of parameters between the client and the TOE. 

TOE SFRs	How the SFR is Met
	Using wildcards is not supported in identity certificates, such as when you import the certificate and private key into ESA. Certificate pinning is also not supported in the evaluated configuration. Since RSA is being used for key exchange and authentication there are no specific parameters associated with the server key exchange. Using the below TLS_RSA ciphers the RSA public key (with a minimum RSA key size 2048) is used for authentication and key exchange. Using the below TLS_DHE ciphers the standard diffie hellman parameters P, Q, and G are used for key exchange. The supported ciphersuites include the following: • TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268 • TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268 • TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268 • TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246 • TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246 • TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256 as defined in RFC 5246 • TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246 Once configured, the TOE will not establish TLS v1.0, SSL2.0 or SSL3.0 connections if offered by the client and only the supported/configured TLS ciphersuites will be used to establish the session. In addition, the TOE will only establish a connection if the peer presents a valid X509 certificate during the handshake.
FIA_AFL.1	The TOE provides the Authorized Administrators the ability to specify the maximum number of unsuccessful authentication attempts before Authorized Administrator is locked out through the administrative CLI and GUI interfaces. While the TOE supports a range from 1-25 with a default of 5 attempts, in the evaluated configuration, the maximum number of failed attempts is required to be set to 3. When the Authorized Administrator attempting to log into the administrative CLI or GUI interface reaches the administratively set maximum number of failed authentication attempts, the user will not be granted access to the administrative functionality of the TOE until an Authorized Administrator resets the user's number of failed login attempts through the administrative CLI using the *userconfig* command or GUI Edit User webpage. The TOE includes the following administrative roles and access: • "admin" default user account that has full access to all system configuration settings. Note, this account is not subject to the lock out at the local console. This is to ensure the administrators do not get totally locked out of the TOE. • "Administrators" have full access to all system configuration settings. This Authorized Administrator account does meet the lockout criteria at the local console and when remotely connected to the TOE via the GUI (secured with HTTPS/TLS) and therefore should be used for the daily management of the TOE.
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 (that include: "!", "@", "#", "$", "%", "^", "&", "*", "(", and ")". By default, the password can be set from 0 to 128 characters, however in the evaluated configuration the password length must be configured to enforce a minimum of 15

TOE SFRs	How the SFR is Met
	characters. The number that is set, is the minimum number of charters that will be required, and the upper limit value can be a range of 15 characters or greater. Refer to the Common Criteria Operational User Guidance and Preparative Procedures for command description and usage information.
FIA_UIA_EXT.1 FIA_UAU_EXT.2	The TOE requires all users to be successfully identified and authenticated before allowing any TSF mediated actions to be performed, except for the login banner that is displayed prior to user authentication. Administrative access to the TOE is facilitated through the TOE's CLI and GUI. The TOE mediates all administrative actions through the CLI and GUI. Once the administrative user attempts to access the CLI via either a directly connected console or remotely through SSHv2, the TOE prompts the user for a user name and password. Likewise, when the administrative user attempts to access the web-based GUI of the TOE through HTTPS over TLSv1.1/2, 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 the Authorized Administrators is successfully identified and authenticated. The TOE provides a local password-based authentication mechanism for the CLI when accessed both locally and remotely as well as the GUI. When the CLI is accessed remotely, the session is secured via SSHv2 and authenticated using SSH public key. The password mechanism can be configured to require passwords to be a minimum of 15 characters from the printable character set. The TOE prevents administrative user actions from being performed prior to successful identification and authentication of the Authorized Administrators. ***Note****,* however, that users accessing the CLI via SSHv2 can be authenticated using public key cryptography. This requires the user's public key to be entered into the TOE (using the *sshconfig* command) and associated with the user's account. If there is no public key configured for the user, the user will instead be prompted to enter a password to authenticate.
FIA_UAU.7	When a user enters their password at the directly connected local console, the TOE will not echo any characters so that the user password is obscured. For remote session authentication via SSHv2 or TLSv1.1/2 secured connection, the TOE does not echo any characters as they are entered.
FIA_X509_EXT.1/Rev FIA_X509_EXT.2 FIA_X509_EXT.3	The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for TLS connections. The certificate validation checking takes place when the certificate is imported. The TOE supports the following methods to obtain a certificate from a CA: • Manual cut-and-paste - ESA generates the Certificate Request Message as described in RFC 2986 which contains the public key and is displayed via the GUI or CLI interface. This allows the administrator to copy the certificate request and in a secure offline manner send the request to a Certification Authority to be transformed into an X.509v3 public-key certificate. • Both the certificate request message and the certificates themselves provide protection in that they are digitally signed. If a certificate is modified in any way, it would be invalidated. The digital signature verifications process would show that the certificate had been tampered with when the

<table>
<tr><th>TOE SFRs</th><th>How the SFR is Met</th></tr>
<tr><td></td><td>hash value would be invalid.
The certificate chain establishes a sequence of trusted certificates, from a peer certificate to the root CA certificate. Within the PKI hierarchy, all enrolled peers can validate the certificate of one another if the peers share a trusted root CA certificate or a common subordinate CA. When a certificate chain is received from a peer, the default processing of a certificate chain path continues until the first trusted certificate is reached.
The Authorized Administrator can also configure one or more certificate fields as listed below that will be used to compare the imported certificate to specific criteria such as:
• alt-subject-name (If subject name doesn't match request, then the alternative subject name filed is used)
• expires-on (If certificate is expired, rejects certificate)
• issuer-name (Is there a trusted root certificate installed for the CA that signed the certificate).
• name (Does the name in the request match the name in the certificate)
• serial-number (Has the certificate been revoked. Serial number will be in the CRL)
• subject-name (Does the name in the request match the name in the certificate)
The administrative user manually installs and selects the certificate used by the TOE for each certificate.

The physical security of the TOE (A.PHYSICAL_PROTECTION) protects ESA and the certificates from being tampered with or deleted. In addition, the TOE identification and authentication security functions protect an unauthorized user from gaining access to the TOE.

The use of CRL is configurable and may be used for certificate revocation. CRL --Certificate checking is performed by a CRL. This is the default option. the TOE performs revocation checking of the entire cert chain (CRL is configured for the certificate authority) at the time of import of the leaf and checks all CA certs (except the trust anchor) every time a leaf is imported.

Checking is also done for the basicConstraints extension and the CA flag to determine whether they are present and set to TRUE. The local certificate that was imported must contain the basic constraints extension with the CA flag set to TRUE, the check also ensure that the key usage extension is present, and the keyEncipherment bit or the keyAgreement bit or both are set. If they are not, the certificate is not accepted.

All the certificates include at least the following information: public key, Common Name, Organization, Organizational Unit and Country.

If the connection to determine the certificate validity cannot be established, ESA will accept the certificate based on the last known state.</td></tr>
<tr><td>FMT_MOF.1/Functions
FMT_MOF./ManualUpdate
FMT_MTD.1/CoreData
FMT_MTD.1/CryptoKeys</td><td>The TOE provides administrative users with a CLI and web-based GUI to interact with and manage the security functions of the TOE. The CLI is the main interface used to administer the TOE since all functionality to configure, securely manage and to monitor the TOE is available via the CLI. The GUI interface can also be used however not all functionality to configure the TOE is available in the GUI. In the evaluated configuration it is recommended to use the CLI to perform all configuration and setting of the security functions and to securely mange the TOE.

No administrative functionality is available prior to the Authorized Administrators logging</td></tr>
</table>

TOE SFRs	How the SFR is Met
	in and being identified and authenticated. Through the CLI, the TOE provides the ability for Authorized Administrators to manage TOE data, such as audit data, configuration settings, cryptographic keys, security attributes, uploading and enabling X509 certificates and login banners via the CLI and GUI. A subset of functionality is available in the GUI. For example, the TOE can initially be installed and set up using the GUI via the System Setup Wizard and saving the config file, selecting the SCP Push method for sending the log files to the remote syslog server and setting inactive timeout. The term "Authorized Administrator" is used in this ST to refer to any user which has beenassigned to a privilege level that is permitted to perform the relevant action; therefore, has the appropriate privileges to perform the requested functions. Therefore, semi-privileged administrators with only a subset of privileges can also modify TOE data based on if granted the privilege. See FMT_SMR.2 for more details on the TOE roles and related privileges. Manual software updates can only be done by the Authorized Administrator through either the CLI or GUI. These updates include software upgrades. The TOE also provides the ability for Authorized Administrators to generate and manage the cryptographic keys that used to secure connections on the TOE. The Authorized Administrators accesses the CLI for management of the cryptographic functions.
FMT_SMF.1	The TOE provides all the capabilities necessary to securely manage the TOE. The Security Administrators (a.k.a Authorized Administrators) user can connect to the TOE using the CLI to perform these functions via SSHv2 secured connection, via the GUI over HTTPS/TLS or at the local console. The CLI is the main interface used to administer the TOE since all functionality to configure, securely manage and to monitor the TOE is available via the CLI. The GUI interface can also be used however not all functionality to configure the TOE is available in the GUI. Therefore, in the evaluated configuration it is recommended to use the CLI to perform all configuration and setting of the security functions and to securely mange the TOE. The specific management capabilities available from the TOE include: • Local and remote administration of the TOE and the services provided by the TOE via the TOE CLI and GUI interfaces, as described above; • The ability to manage the warning banner message and content which allows the Authorized Administrator the ability to define warning banner that is displayed prior to establishing a session (note this applies to the interactive (human) users; e.g. administrative users; • The ability to manage the time limits of session inactivity which allows the Authorized Administrator the ability to set and modify the inactivity time threshold; • The ability to configure the number of failed administrator logon attempts that will cause the account to be locked until it is reset; • The ability to re-enable an administrator's account that has been locked; • The ability to update the AsyncOS software. The validity of the image is provided using SHA-384 hash prior to installing the update; • The ability to manage audit behavior and the audit logs which allows the Authorized Administrator to configure the audit logs, view the audit logs, and to clear the audit logs; • The ability to manage the cryptographic functionality which allows the Authorized Administrator the ability to identify and configure the algorithms used to provide

TOE SFRs	How the SFR is Met
	protection of the data, such as generating the RSA keys to enable SSHv2 and TLSv1.1/2; • The ability to configure the SSHv2 functionality which supports the secure connections to the audit server; • The ability to import the X.509v3 certificates and validate for use in authentication and secure connections; • The ability to configure and set the time clock. A subset of functionality is available in the GUI. For example, the TOE can initially be installed and set up using the GUI via the System Setup Wizard and saving the config file, selecting the SCP Push method for sending the log files to the remote syslog server and setting inactive timeout.
FMT_SMR.2	The TOE maintains Authorized Administrators that include privileged and semi-privileged administrator roles to administer the TOE locally and remotely. The term s "Authorized Administrator" and "Security Administrator" may be used interchangeable in this ST to refer to any user that has been assigned to a privilege level that is permitted to perform the relevant action; therefore, has the appropriate privileges to perform the requested functions. The assigned role determines the functions the user can perform; hence the Authorized Administrator with the appropriate privileges. The TOE performs role-based authorization, using TOE platform authorization mechanisms, to grant access to the semi-privileged and privileged roles. The default user account for ESA is 'admin' and has all administrative privileges. The admin user account cannot be deleted, but an Authorized Administrator can change the password and lock the account, which is recommended. When an Authorized Administrator creates a new user account, they can assign the user to a predefined or a custom user role. Each role contains differing levels of permissions within the system. Although there is no limit to the number of user accounts that an Authorized Administrator can create on the appliance, Authorized Administrator cannot create user accounts with names that are reserved by the system such as "operator" or "root." The following roles are predefined by the system and can be assigned to user accounts: • admin - default user account that has full access to all system configuration settings. • Administrator - has full access to all system configuration settings. • Technician - can perform system upgrades, reboot the appliance, and manage key features. • Operators - are restricted from creating, editing, or removing user accounts and cannot use the following commands: resetconfig, upgradecheck, upgradeinstall, systemsetup or running the System Setup Wizard. • Read-Only Operator - can view administrative interfaces, but do not have the ability to commit configuration changes or to access the file system or SCP, thus preventing them from accessing log files • Guest - can only view system status information. The term "Authorized Administrator" is used in this ST to refer to any user which has been assigned to a privilege level that is permitted to perform the relevant action; therefore, has the appropriate privileges to perform the requested functions. The privilege level determines the functions the user can perform; hence the Authorized Administrator with the appropriate privileges.

TOE SFRs	How the SFR is Met
	The TOE can and shall be configured to authenticate all access to the CLI and GUI using a username and password. The TOE supports both local administration via a directly connected console cable and remote administration via CLI using SSHv2 and via the GUI using HTTPS/TLS secure connection.
FPT_SKP_EXT.1 and FPT_APW_EXT.1	In the evaluated configuration, the TOE must run in FIPS mode. To be in FIPS mode, the Authorized Administrator enters the 'fipsconfig' command at the CLI. During the FIPS mode setup, an Authorized Administrator is able to select the option to have all passwords and keys encrypted using AES256-CBC. In addition, there is a sub-option using the 'saveconfig' command and the save config dialog in the GUI to encrypt the passwords and keys. In the evaluated configuration, these options must be selected and configured as described in the Cisco Email Server Appliance (ESA) Common Criteria Operational User Guidance And Preparative Procedures. The encrypted passwords and keys are stored in their respective configuration files and there are no administrative interfaces available to access the data. Refer to the Common Criteria Operational User Guidance and Preparative Procedures for command description and usage information.
FPT_STM_EXT.1	The TOE provides a source of date and time information used in audit event timestamps. The clock function is reliant on the system clock provided by the underlying hardware in the physical TOE devices and synchronizing time with the ESXi hypervisor in the case of virtual TOE devices.. This date and time is used as the time stamp that is applied to TOE generated audit records and used to track inactivity of administrative sessions. The time information is also used in setting the system time and administrative session timeout. The time can be configured using the CLI commands: settime and settz. In the GUI, the time can be configured under the Time Zone or Time Settings page from the System Administration menu.
FPT_TUD_EXT.1	An Authorized Administrator can query the currently executing software version via the CLI and GUI. An Authorized Administrator can either manually downloads the updates or ESA can automatically download the updates when "automated updates" has been configured. Note, in the evaluated configuration, automated updates will not be allowed. Updates can be downloaded directly from the Cisco Update Servers as well as from an offline update server. Both an Authorized Administrator and the TOE can check to see if an update is available from Cisco. The Authorized Administrator can also verify the downloaded SHA384 hash. Once the file is downloaded to a server, the Authorized Administrator verifies that it was not tampered with prior to moving it to the TOE by using a SHA-384 utility to compute an SHA-384 hash for the downloaded file and comparing this with the SHA-384 hash for the image listed on the download page on Cisco.com. Once the Authorized Administrator has verified the TOE image, the file can be installed. Attempts to perform an illegitimate update onto the system will be logged into updater logs at INFO level. The sample log line will look as follows:

TOE SFRs	How the SFR is Met
	Wed Dec 11 05:50:07 2013 Info: repeng SHA384 Mismatch If the there is an issue with the verification of the SHA384 checksum, the software should not be installed and the Authorized Administrator contacts Cisco TAC for assistance. For full details, refer to the Cisco Email Server Appliance (ESA) Common Criteria Operational User Guidance And Preparative Procedures for assistance.
FPT_TST_EXT.1	During the system bootup process (power on or reboot), all the Power on Startup Test (POST) components for all the cryptographic modules perform the POST for the corresponding component (hardware or software). Also, during the initialization and self-tests, the module inhibits all access to the cryptographic algorithms. Additionally, the power-on self-tests are performed after the cryptographic systems are initialized but prior to the underlying OS initialization of external interfaces; this prevents the security appliances from passing any data before completing self-tests and entering FIPS mode. In the event of a power-on self-test failure of any component, the system crashes and appropriate information is displayed on the screen, and an alert is sent to an administrative email each time a self-test fails for any reason and a failed part of the functionality is disabled until a problem resolution has been accomplished This operation ensures no cryptographic algorithms can be accessed unless all power on self-tests are successful. The tests include: • AES Known Answer Test – For the encrypt test, a known key is used to encrypt a known plain text value resulting in an encrypted value. This encrypted value is compared to a known encrypted value to ensure that the encrypt operation is working correctly. The decrypt test is just the opposite. In this test a known key is used to decrypt a known encrypted value. The resulting plaintext value is compared to a known plaintext value to ensure that the decrypt operation is working correctly. • RSA Signature Known Answer Test (both signature/verification) – This test takes a known plaintext value and Private/Public key pair and used the public key to encrypt the data. This value is compared to a known encrypted value to verify that encrypt operation is working properly. The encrypted data is then decrypted using the private key. This value is compared to the original plaintext value to ensure the decrypt operation is working properly. • RNG/DRBG Known Answer Test – For this test, known seed values are provided to the DRBG implementation. The DRBG uses these values to generate random bits. These random bits are compared to known random bits to ensure that the DRBG is operating correctly. • HMAC Known Answer Test – For each of the hash values listed, the HMAC implementation is fed known plaintext data and a known key. These values are used to generate a MAC. This MAC is compared to a known MAC to verify that the HMAC and hash operations are operating correctly. • SHA-1/256/512 Known Answer Test – For each of the values listed, the SHA implementation is fed known data and key. These values are used to generate a hash. This hash is compared to a known value to verify they match and the hash operations are operating correctly. Prior to installing the image, the Authorized Administrator can verify the public hash to ensure the files has not been tampered with prior to installing. Using a SHA-384 utility, the

TOE SFRs	How the SFR is Met
	Authorized Administrator can compute a SHA-384 hash for the downloaded file and compare the results with the SHA-384 hash on the Cisco.com download page. The Software Integrity Test is run automatically whenever the AsyncOS system images is loaded and confirms that the image file that's about to be loaded has maintained its integrity with the signature verification of the file image. The Software Integrity Test is also run automatically whenever the AsyncOS system is rebooted. The FOM cryptographic module that is part of the TOE image, performs both power-up self-tests at Module initialization and continuous conditional tests during operation. Input, output, and cryptographic functions cannot be performed while the Module is in a self-test or error state as the Module is single threaded and will not return to the calling application until the power-up self-tests are complete. If the power-up self-tests fail subsequent calls to the Module will fail and thus no further cryptographic operations are possible. Additionally, within the system, /etc/rc.d/init.d/verify_fsic calls verify_file_integ.sh which extracts, validates and merges hash databases generated and signed at build time. For each file in the database a current hash is calculated and compared to the hash recorded in the database. If any of the cryptographic tests or comparison of the hash values fail, the TOE will enter an error state or reboot in attempts to correct the problem. If the issue is not resolved, the Authorized Administrator contacts Cisco TAC for assistance. If any component reports failure for the POST, the system crashes and appropriate information is displayed on the screen, and an alert is sent to an administrative email each time a self-test fails for any reason and a failed part of the functionality is disabled until a problem resolution has been accomplished. All ports are blocked from moving to forwarding state during the POST. If all components of all modules pass the POST, the system is placed in FIPS PASS state and ports are allowed to forward data traffic. These tests are sufficient to verify that the correct version of the TOE software is running as well as that the cryptographic operations are all performing as expected because any deviation in the TSF behaviour will be identified by the failure of a self-test.
FTA_SSL_EXT.1 and FTA_SSL.3	The Authorized Administrators can configure maximum inactivity times individually for both the CLI and GUI. The Authorized Administrator can specify how long a user can be logged into the GUI before the user is logged out due to inactivity by default it is set to 30 minutes. Once AsyncOS logs a user out, the appliance redirects the user's web browser to the login page. Likewise, the Authorized Administrator can specify how long a user can be logged into the Email Security appliance's CLI before AsyncOS logs the user out due to inactivity. If a local user session is inactive for a configured period of time, the session will be terminated and will require be re-identification and re-authentication to re-establish a new session and access the TOE. If a remote user session is inactive for a configured period of time, the session will be terminated and will require re-identification and re-authentication to establish a new session and access the TOE.
FTA_SSL.4	An administrator is able to exit out of both the CLI and GUI administrative sessions. The Authorized Administrator can log out of the CLI with the 'exit' command. The Web UI also has a logout option via the drop-down menu
FTA_TAB.1	The Authorized Administrator defines a custom login banner that will be displayed at the GUI and the CLI for both local and remote access configurations prior to allowing Authorized Administrator access through those interfaces.

TOE SFRs	How the SFR is Met
	A local console includes any IT Environment Console that is directly connected to the TOE via the Serial Console Port and is used by the Authorized Administrator to support TOE administration. Whereas a remote console is one that includes any IT Environment Management workstation with one of the supported Web Browsers or any SSH client that supports SSHv2 may be used by the Authorized Administrator to support TOE administration through HTTPS/TLS or SSH protected channels.
FTP_ITC.1	The TOE protects communications with the syslog server using SCP over SSHv2. SSHv2 uses a keyed hash as defined in FCS_SSHC_EXT.1.6. This protects the data from modification by hashing the data and verifying the hash on receipt of the data. This ensures that the data has not been modified in transit. In addition, encryption of the data as defined in FCS_SSHC_EXT.1.4 is provided to ensure the data is not disclosed in transit. SCP Push is used for sending audit logs securely over SSHv2 to a syslog server. This method periodically pushes log files to a remote Syslog server. It requires an SSH server on the Syslog Server using the SSHv2 protocol. The subscription requires a username, SSH key, and destination directory on the remote computer. Log files are transferred based on a rollover schedule set by an Authorized Administrator.
FTP_TRP.1/Admin	All remote administrative communications take place over a secure encrypted SSHv2 for the CLI or TLS/HTTPS for the GUI sessions. The SSHv2 session is encrypted using AES encryption. The remote users are able to initiate SSHv2 communications with the TOE for secure CLI access. TLS/HTTPS is used to secure the communications with the TOE and remote web browser for secure GUI access.

7 ANNEX A: KEY ZEROIZATION

7.1 Key Zeroization

The following table describes the key zeroization referenced by FCS_CKM.4 provided by the TOE. As described below in the table, the TOE zeroize all secrets, keys and associated values when they are no longer required. The process in which the TOE zeroizes, meets FIPS 140 validation.

Table 20 TOE Key Zeroization

Name	Description	Stored	Zeroization
Diffie-Hellman Shared Secret	The value is zeroized after it has been given back to the consuming operation. The value is overwritten by 0's.	This key is stored in DRAM.	Automatically after completion of DH exchange. Overwritten with: 0x00
Diffie Hellman private exponent	This is the private exponent used as part of the Diffie-Hellman key exchange.	This key is stored in DRAM.	Zeroized upon completion of DH exchange. Overwritten with: 0x00
SSH Private Key	Once the function has completed the operations requiring the RSA key object, the module over writes the entire object (no matter its contents).	This key is stored in NVRAM	Zeroized upon deletion of the SSH public/private key pair when no longer needed Overwritten with: 0x00
SSH Session Key	Once the function has completed the operations requiring the RSA key object, the module over writes the entire object (no matter its contents).	This key is stored in DRAM.	Automatically when the SSH session is terminated. Overwritten with: 0x00
TLS server private key	This key is used for authentication, so the server can prove who it is. The private key used for TLS secure connections.	This key is stored in NVRAM.	Zeroized by overwriting with new key
TLS server public key	This key is used to encrypt the data that is used to compute the secret key. The public key used for TLS secure connection.	This key is stored in NVRAM.	Zeroized by overwriting with new key
TLS pre-master secret	The pre-master secret is the client and server exchange of random numbers and a special number, the pre-master secret, this pre-master secret is using asymmetric cryptography from which new TLS session keys can be created.	This key is stored in SDRAM.	Automatically after TLS session terminated. The value is overwritten with "0x00."
TLS session encryption key	The session encryption key is unique for each session and is based on the shared secrets that were negotiated at the start of the session. The Key is used to encrypt TLS session data.	This key is stored in SDRAM.	Automatically after TLS session terminated. The value is overwritten with "0x00."
TLS session integrity key	This key is used to provide the privacy and TLS data integrity protection.	This key is stored in SDRAM.	Automatically after TLS session terminated. The entire object is overwritten with zeros
User Password	This is a variable 15+ character password that is used to authenticate local users.	The password is stored in NVRAM.	Zeroized by overwriting with new password

8 ANNEX B: REFERENCES

The following documentation was used to prepare this ST:

Table 21 References

Identifier	Description
[CC_PART1]	Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and general model, Version 3.1, Revision 5, dated: April 2017
[CC_PART2]	Common Criteria for Information Technology Security Evaluation – Part 2: Security functional components, Version 3.1, Revision 5, dated: April 2017
[CC_PART3]	Common Criteria for Information Technology Security Evaluation – Part 3: Security assurance components, Version 3.1, Revision 5, dated: April 2017
[CEM]	Common Methodology for Information Technology Security Evaluation – Evaluation Methodology, Version 3.1, Revision 5, dated: April 2017
[NDcPP]	collaborative Protection Profile for Network Devices + Errata 20180314, Version 2.1, 24 September 2018
[800-56A]	NIST Special Publication 800-56A, March, 2007
[800-56B]	NIST Special Publication 800-56B Recommendation for Pair-Wise, August 2009
[FIPS 140-2]	FIPS PUB 140-2 Federal Information Processing Standards Publication
[FIPS PUB 186-3]	FIPS PUB 186-3 Federal Information Processing Standards Publication Digital Signature Standard (DSS) June, 2009
[800-90]	NIST Special Publication 800-90A Recommendation for Random Number Generation Using Deterministic Random Bit Generators January 2012
[FIPS PUB 180-3]	FIPS PUB 180-3 Federal Information Processing Standards Publication Secure Hash Standard (SHS) October 2008

9 ANNEX C: EXTENDED COMPONENTS DEFINITIONS

The NDcPPv2.1 Author has defined extended components that are claimed in this Security Target (ST). Extended SFRs are identified by having a label "EXT" at the end of the Security Functional Requirement name.

The following are the extended components claimed in this ST:

Table 22 Extended Components

Component Identification	Component Name
FAU_STG_EXT.1	Protected Audit Event Storage
FCS_RBG_EXT.1	Cryptographic Operation (Random Bit Generation)
FCS_SSHC_EXT.1	SSH Client Protocol
FCS_SSHS_EXT.1	SSH Server Protocol
FCS_TLSS_EXT.1	TLS Server Protocol
FCS_RBG_EXT.1	Cryptographic Operation (Random Bit Generation)
FIA_PMG_EXT.1	Password Management
FIA_UIA_EXT.1	User Identification and Authentication
FIA_UAU_EXT.2	Password-based Authentication Mechanism
FIA_X509_EXT.1/Rev	X.509 Certificate Validation
FIA_X509_EXT.2	X.509 Certificate Authentication
FIA_X509_EXT.3	X.509 Certificate Requests
FPT_APW_EXT.1	Protection of Administrator Passwords
FPT_SKP_EXT.1	Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)
FPT_STM_EXT.1	Reliable Time Stamps
FPT_TST_EXT.1	TSF Testing
FPT_TUD_EXT.1	Trusted Update
FTA_SSL_EXT.1	TSF-initiated Session Locking

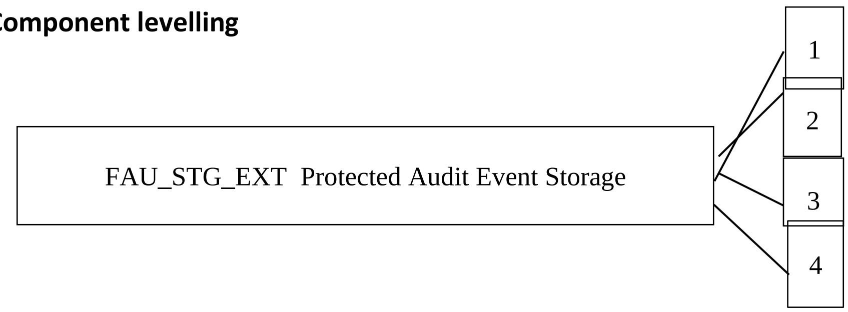
9.1 Security Audit (FAU)

9.1.1 Protected audit event storage (FAU_STG_EXT)

Family Behaviour

This component defines the requirements for the TSF to be able to securely transmit audit data between the TOE and an external IT entity.

Component levelling



FAU_STG_EXT.1 Protected audit event storage requires the TSF to use a trusted channel implementing a secure protocol.

FAU_STG_EXT.2 Counting lost audit data requires the TSF to provide information about audit records affected when the audit log becomes full.

FAU_STG_EXT.3 Protected Local audit event storage for distributed TOEs requires the TSF to use a trusted channel to protect audit transfer to another TOE component.

FAU_STG_EXT.4 Protected Remote audit event storage for distributed TOEs requires the TSF to use a trusted channel to protect audit transfer to another TOE component.

Management: FAU_STG_EXT.1, FAU_STG_EXT.2, FAU_STG_EXT.3, FAU_STG_EXT.4

The following actions could be considered for the management functions in FMT:

a) The TSF shall have the ability to configure the cryptographic functionality.

Audit: FAU_STG_EXT.1, FAU_STG_EXT.2, FAU_STG_EXT.3, FAU_STG_EXT.4

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) No audit necessary.

9.1.1.1 FAU_ STG_EXT.1 Protected Audit Event Storage

FAU_STG_EXT.1 Protected Audit Event Storage

Hierarchical to: No other components.

Dependencies: FAU_GEN.1 Audit data generation
FTP_ITC.1 Inter-TSF Trusted Channel

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.

Application Note 136

For selecting the option of transmission of generated audit data to an external IT entity the TOE relies on a non-TOE audit server for storage and review of audit records. The storage of these audit records and the ability to allow the Administrator to review these audit records is provided by the operational environment in that case. Since the external audit server is not part of the TOE, there are no requirements on it except the capabilities for ITC transport for audit data. No requirements are placed upon the format or underlying protocol of the audit data being transferred. The TOE must be capable of being configured to transfer audit data to an external IT entity without Administrator intervention. Manual transfer would not meet the requirements. Transmission could be done in real-time or periodically. If the transmission is not done in real-time then the TSS describes what event stimulates the transmission to be made and what range of frequencies the TOE supports for making transfers of audit data to the audit server; the TSS also suggests typical acceptable frequencies for the transfer.

For distributed TOEs each component must be able to export audit data across a protected channel external (FTP_ITC.1) or intercomponent (FPT_ITT.1 or FTP_ITC.1) as appropriate. At least one component of the TOE must be able to export audit records via FTP_ITC.1 such that all TOE audit records can be exported to an external IT entity.

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

- *TOE shall consist of a single standalone component that stores audit data locally,*
- *The TOE shall be a distributed TOE that stores audit data on the following TOE components: [assignment: identification of TOE components],*
- *The TOE shall be a distributed TOE with storage of audit data provided externally for the following TOE components: [assignment: list of TOE components that do not store audit data locally and the other TOE components to which they transmit their generated audit data]*.

Application Note 137

If the TOE is a standalone TOE (i.e. not a distributed TOE) the option 'The TOE shall consist of a single standalone component that stores audit data locally' shall be selected.

If the TOE is a distributed TOE the option 'The TOE shall be a distributed TOE that stores audit data on the following TOE components: [assignment: identification of TOE components]' shall be selected and the TOE components which store audit data locally shall be listed in the assignment. Since all TOEs are required to provide functions to store audit data locally this option needs to be selected for all distributed TOEs. In addition, FAU_GEN_EXT.1 and FAU_STG_EXT.3 shall be claimed in the ST. If the distributed TOE consists only of components which are storing audit data locally, it is sufficient to select only the option 'The TOE shall be *a distributed TOE that stores audit data on the following TOE components: [assignment: identification of TOE components]' and add FAU_GEN_EXT.1 and FAU_STG_EXT.3.*

If the TOE is a distributed TOE and some TOE components are not storing audit data locally, the option 'The TOE shall be a distributed TOE with storage of audit data provided externally for the following TOE components: [assignment: list of TOE components that do not store audit data locally and the other TOE components to which they transmit their generated audit data]' shall be selected in addition to the option 'The TOE shall be a distributed TOE that stores audit data on the following TOE components: [assignment: identification of TOE components]'. In that case FAU_STG_EXT.4 shall be claimed in the ST in addition to FAU_GEN_EXT.1 and FAU_STG_EXT.3. For the option 'The TOE shall be a distributed TOE with storage of audit data provided externally for the following TOE components: [assignment: list of TOE components that do not store audit data locally and the other TOE components to which they transmit their generated audit data]' the TOE components that to not store audit data locally shall be mapped to the TOE components to which they transmit their generated audit data.

For distributed TOEs this SFR can be fulfilled either by every TOE component storing its own security audit data locally or by one or more TOE components storing audit data locally and other TOE components which are not storing audit information locally sending security audit data to other TOE components for local storage. For the transfer of security audit data between TOE components a protected channel according to FTP_ITC.1 or FPT_ITT.1 shall be used. The TSS shall describe which TOE components store security audit data locally and which TOE components do not store security audit data locally. For the latter, the TSS shall describe at which other TOE component the audit data is stored locally.

FAU_STG_EXT.1.3 The TSF shall [selection: *drop new audit data, overwrite previous audit records according to the following rule: [assignment: rule for overwriting previous audit records], [assignment: other action]*] when the local storage space for audit data is full.

Application Note 138

The external log server might be used as alternative storage space in case the local storage space is full. The "other action" could in this case be defined as "send the new audit data to an external IT entity".

For distributed TOEs each component is not required to store generated audit data locally but the overall TOE needs to be able to store audit data locally. Each component must at least provide the ability to temporarily buffer audit information locally to ensure that audit records are preserved in case of network connectivity issues. Buffering audit information locally, does not necessarily involve non-volatile memory: audit information could be buffered in volatile memory. However, the local storage of audit information in the sense of FAU_STG_EXT.1.3 needs to be done in non-volatile memory. For every component which performs local storage of audit information, the behaviour when local storage is exhausted needs to be described. For every component which is buffering audit information instead of storing audit information locally itself, it needs to be described what happens in case the buffer space is exhausted.

9.2 Cryptographic Support (FCS)

9.2.1 Random Bit Generation (FCS_RBG_EXT)

9.2.1.1 FCS_RBG_EXT.1 Random Bit Generation

Family Behaviour

Components in this family address the requirements for random bit/number generation. This is a new family defined for the FCS class.

Component levelling

FCS_RBG_EXT.1 Random Bit Generation requires random bit generation to be performed in accordance with selected standards and seeded by an entropy source.

Management: FCS_RBG_EXT.1

The following actions could be considered for the management functions in FMT:

a) There are no management activities foreseen

Audit: FCS_RBG_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) Minimal: failure of the randomization process

9.2.1.2 FCS_RBG_EXT.1 Random Bit Generation

Hierarchical to: No other components
Dependencies: No other components

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

FCS_RBG_EXT.1.2 The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [selection: *[assignment: number of software-based sources] software-based noise source, [assignment: number of hardware-based sources] hardware-based noise source*] with a minimum of [selection: *128 bits, 192 bits, 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.

Application Note 142

For the first selection in FCS_RBG_EXT.1.2, the ST author selects at least one of the types of noise sources. If the TOE contains multiple noise sources of the same type, the ST author fills the assignment with the *appropriate number for each type of source (e.g., 2 software-based noise sources, 1 hardware-based noise source). The documentation and tests required in the Evaluation Activity for this element should be repeated to cover each source indicated in the ST.*

ISO/IEC 18031:2011 contains three different methods of generating random numbers; each of these, in turn, depends on underlying cryptographic primitives (hash functions/ciphers). The ST author will select the function used and include the specific underlying cryptographic primitives used in the requirement. While any of the identified hash functions (SHA-1, SHA-256, SHA-384, SHA-512) are allowed for Hash_DRBG or HMAC_DRBG, only AES-based implementations for CTR_DRBG are allowed.

If the key length for the AES implementation used here is different than that used to encrypt the user data, then FCS_COP.1 may have to be adjusted or iterated to reflect the different key length. For the selection in FCS_RBG_EXT.1.2, the ST author selects the minimum number of bits of entropy that is used to seed the RBG, which must be equal or greater than the security strength of any key generated by the TOE.

9.2.2 Cryptographic Protocols (FCS_DTLSC_EXT, FCS_DTLSS_EXT, FCS_HTTPS_EXT, FCS_IPSEC_EXT, FCS_NTP_EXT, FCS_SSHC_EXT, FCS_SSHS_EXT, FCS_TLSC_EXT, FCS_TLSS_EXT)

9.2.2.1 FCS_HTTPS_EXT.1 HTTPS Protocol

Family Behaviour

Components in this family define the requirements for protecting remote management sessions between the TOE and a Security Administrator. This family describes how HTTPS will be implemented. This is a new family defined for the FCS Class.

Component levelling

FCS_HTTPS_EXT.1 HTTPS requires that HTTPS be implemented according to RFC 2818 and supports TLS.

Management: FCS_HTTPS_EXT.1

The following actions could be considered for the management functions in FMT:

a) There are no management activities foreseen.

Audit: FCS_HTTPS_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) There are no auditable events foreseen.

9.2.2.2 FCS_HTTPS_EXT.1 HTTPS Protocol

Hierarchical to: No other components
Dependencies: [FCS_TLSC_EXT.1 TLS Client Protocol, or
FCS_TLSS_EXT.1 TLS Server 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 the HTTPS protocol using TLS.

FCS_HTTPS_EXT.1.3 If a peer certificate is presented, the TSF shall [selection: *not establish the connection, request authorization to establish the connection, [assignment: other action]*] if the peer certificate is deemed invalid.

9.2.2.3 FCS_SSHC_EXT.1 SSH Client

Family Behaviour

The component in this family addresses the ability for a client to use SSH to protect data between the client and a server using the SSH protocol.

Component levelling

FCS_SSHC_EXT.1 SSH Client requires that the client side of SSH be implemented as specified.

Management: FCS_SSHC_EXT.1

The following actions could be considered for the management functions in FMT:

There are no management activities foreseen.

Audit: FCS_SSHC_EXT.1

The following actions should be considered for audit if FAU_GEN Security audit data generation is included in the PP/ST:

a) Failure of SSH session establishment
b) SSH session establishment
c) SSH session termination

9.2.2.4 FCS_SSHC_EXT.1 SSH Client Protocol

Hierarchical to: No other components

Dependencies: FCS_CKM.1Cryptographic Key Generation
FCS_CKM.2 Cryptographic Key Establishment
FCS_COP.1/DataEncryption Cryptographic operation (AES Data encryption/decryption)
FCS_COP.1/SigGen Cryptographic operation (Signature Generation and Verification)
FCS_COP.1/Hash Cryptographic operation (Hash Algorithm)
FCS_COP.1/KeyedHash Cryptographic operation (Keyed Hash Algorithm)
FCS_RBG_EXT.1 Random Bit Generation

FCS_SSHC_EXT.1.1 The TSF shall implement the SSH protocol that complies with RFC(s) [selection: *4251, 4252, 4253, 4254, 5647, 5656, 6187, 6668, 8332*].

Application Note 178

The ST author selects which of the RFCs to which conformance is being claimed. Note that these need to be consistent with selections in later elements of this component (e.g., cryptographic algorithms permitted). RFC 4253 indicates that certain cryptographic algorithms are "REQUIRED". This means that the implementation must include support, not that the algorithms must be enabled for use. Ensuring that algorithms indicated as "REQUIRED" but not listed in the later elements of this component are implemented is out of scope of the evaluation activity for this requirement.

RFC 5647 only applies to the RFC compliant implementation of GCM; a TOE that only implements the "@openssh.com" variant of GCM should not select 5647. aes-gcm@openssh.com is specified in Section 1.6 of the OpenSSH Protocol Specification (https://cvsweb.openbsd.org/cgi-bin/cvsweb/src/usr.bin/ssh/PROTOCOL?rev=1.31).*

FCS_SSHC_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, [selection: *password-based, no other method*].

FCS_SSHC_EXT.1.3 The TSF shall ensure that, as described in RFC 4253, packets greater than [assignment: *number of bytes*] bytes in an SSH transport connection are dropped.

Application Note 173

RFC 4253 provides for the acceptance of "large packets" with the caveat that the packets should be of "reasonable length" or dropped. The assignment should be filled in by the ST author with the maximum packet size accepted, thus defining "reasonable length" for the TOE.

FCS_SSHC_EXT.1.4 The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms and rejects all other encryption algorithms: [assignment: *list of encryption algorithms*].

FCS_SSHC_EXT.1.5 The TSF shall ensure that the SSH public-key based authentication implementation uses [selection: ssh-rsa, ***rsa-sha2-256, rsa-sha2-512***, ecdsa-sha2-nistp256, x509v3-ssh-rsa, ecdsa-sha2-nistp384, ecdsa-sha2-nistp521, x509v3-ecdsa-sha2-nistp256, x509v3-ecdsa-sha2-nistp384, x509v3-ecdsa-sha2-nistp521, x509v3-rsa2048-sha256] as its public key algorithm(s) and rejects all other public key algorithms

FCS_SSHC_EXT.1.6 The TSF shall ensure that the SSH transport implementation uses [assignment: *list of data integrity MAC algorithms*] as its data integrity MAC algorithm(s) and rejects all other MAC algorithm(s).

FCS_SSHC_EXT.1.7 The TSF shall ensure that [assignment: *list of key exchange methods*] are the only allowed key exchange methods used for the SSH protocol.

FCS_SSHC_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 no more than one gigabyte of transmitted data. After either of the thresholds are reached a rekey needs to be performed.

Application Note 180

This SFR defines two thresholds - one for the maximum time span the same session keys can be used and the other one for the maximum amount of data that can be transmitted using the same session keys. Both thresholds need to be implemented and a rekey needs to be performed on whichever threshold is reached first. For the maximum transmitted data threshold, the total incoming and outgoing data needs to be counted. The rekey applies to all session keys (encryption, integrity protection) for incoming and outgoing traffic.

It is acceptable for a TOE to implement lower thresholds than the maximum values defined in the SFR.

For any configurable threshold related to this requirement the guidance documentation needs to specify how the threshold can be configured. The allowed values must either be specified in the guidance documentation and must be lower or equal to the thresholds specified in this SFR or the TOE must not accept values beyond the thresholds specified in this SFR.

FCS_SSHC_EXT.1.9 The TSF shall ensure that the SSH client authenticates the identity of the SSH server using a local database associating each host name with its corresponding public key or [selection: *a list of trusted certification authorities, no other methods*] as described in RFC 4251 section 4.1.

Application Note 181

The list of trusted certification authorities can only be selected if x509v3-ssh-rsa, x509v3-ecdsa-sha2-nistp256, x509v3-ecdsa-sha2-nistp384, x509v3-ecdsa-sha2-nistp521 or x509v3-rsa2048-sha256 are selected in FCS_SSHC_EXT.1.5.

9.2.2.5 FCS_SSHS_EXT.1 SSH Server Protocol

Family Behaviour

The component in this family addresses the ability for a server to offer SSH to protect data between a client and the server using the SSH protocol.

Component levelling

FCS_SSHS_EXT.1 SSH Server requires that the server side of SSH be implemented as specified.

Management: FCS_SSHS_EXT.1

The following actions could be considered for the management functions in FMT:

a) There are no management activities foreseen.

Audit: FCS_SSHS_EXT.1

The following actions should be considered for audit if FAU_GEN Security audit data generation is included in the PP/ST:

a) Failure of SSH session establishment
b) SSH session establishment
c) SSH session termination

Hierarchical to: No other components

9.2.2.6 FCS_SSHS_EXT.1 SSH Server Protocol

Dependencies: FCS_CKM.1Cryptographic Key Generation
FCS_CKM.2 Cryptographic Key Establishment
FCS_COP.1/DataEncryption Cryptographic operation (AES Data encryption/decryption)
FCS_COP.1/SigGen Cryptographic operation (Signature Generation and Verification)
FCS_COP.1/Hash Cryptographic operation (Hash Algorithm)
FCS_COP.1/KeyedHash Cryptographic operation (Keyed Hash Algorithm)
FCS_RBG_EXT.1 Random Bit Generation

FCS_SSHS_EXT.1.1 The TSF shall implement the SSH protocol that complies with RFC(s) [selection: *4251, 4252, 4253, 4254, 5647, 5656, 6187, 6668, 8332*].

Application Note 182

The ST author selects which of the RFCs to which conformance is being claimed. Note that these need to be consistent with selections in later elements of this component (e.g., cryptographic algorithms permitted). RFC 4253 indicates that certain cryptographic algorithms are "REQUIRED". This means that the implementation must include support, not that the algorithms must be enabled for use. Ensuring that algorithms indicated as "REQUIRED" but not listed in the later elements of this component are implemented is out of scope of the evaluation activity for this requirement.

RFC 5647 only applies to the RFC compliant implementation of GCM; a TOE that only implements the "@openssh.com" variant of GCM should not select 5647. aes-gcm@openssh.com is specified in Section 1.6 of the OpenSSH Protocol Specification (https://cvsweb.openbsd.org/cgi-bin/cvsweb/src/usr.bin/ssh/PROTOCOL?rev=1.31).*

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 [assignment: *number of bytes*] bytes in an SSH transport connection are dropped.

Application Note 183

RFC 4253 provides for the acceptance of "large packets" with the caveat that the packets should be of "reasonable length" or dropped. The assignment should be filled in by the ST author with the maximum packet size accepted, thus defining "reasonable length" for the TOE.

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: [assignment: *encryption algorithms*].

FCS_SSHS_EXT.1.5 The TSF shall ensure that the SSH public-key based authentication implementation uses [selection: ssh-rsa, ***rsa-sha2-256, rsa-sha2-512***, ecdsa-sha2-nistp256, x509v3-ssh-rsa, ecdsa-sha2-

nistp384, ecdsa-sha2-nistp521, x509v3-ecdsa-sha2-nistp256, x509v3-ecdsa-sha2-nistp384, x509v3-ecdsa-sha2-nistp521, x509v3-rsa2048-sha256] 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 [assignment: *list of MAC algorithms*] as its MAC algorithm(s) and rejects all other MAC algorithm(s).

FCS_SSHS_EXT.1.7 The TSF shall ensure that [assignment: *list of key exchange 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 no more than one gigabyte of transmitted data. After either of the thresholds are reached a rekey needs to be performed.

Application Note 184

This SFR defines two thresholds - one for the maximum time span the same session keys can be used and the other one for the maximum amount of data that can be transmitted using the same session keys. Both thresholds need to be implemented and a rekey needs to be performed on whichever threshold is reached first. For the maximum transmitted data threshold, the total incoming and outgoing data needs to be counted. The rekey applies to all session keys (encryption, integrity protection) for incoming and outgoing traffic.

It is acceptable for a TOE to implement lower thresholds than the maximum values defined in the SFR.

For any configurable threshold related to this requirement the guidance documentation needs to specify how the threshold can be configured. The allowed values must either be specified in the guidance documentation and must be lower or equal to the thresholds specified in this SFR or the TOE must not accept values beyond the thresholds specified in this SFR.

9.2.2.7 FCS_TLSS_EXT TLS Server Protocol

Family Behaviour

The component in this family addresses the ability for a server to use TLS to protect data between a client and the server using the TLS protocol.

Component levelling

FCS_TLSS_EXT.1 TLS Server requires that the server side of TLS be implemented as specified.

FCS_TLSS_EXT.2: TLS Server requires the mutual authentication be included in the TLS implementation.

Management: FCS_TLSS_EXT.1, FCS_TLSS_EXT.2

The following actions could be considered for the management functions in FMT:

a) There are no management activities foreseen.

Audit: FCS_TLSS_EXT.1, FCS_TLSS_EXT.2

The following actions should be considered for audit if FAU_GEN Security audit data generation is included in the PP/ST:

a) Failure of TLS session establishment
b) TLS session establishment
c) TLS session termination

Hierarchical to: No other components

9.2.2.8 FCS_TLSS_EXT.1 TLS Server Protocol

Dependencies: FCS_CKM.1 Cryptographic Key Generation
FCS_CKM.2 Cryptographic Key Establishment
FCS_COP.1/DataEncryption Cryptographic operation (AES Data encryption/decryption)
FCS_COP.1/SigGen Cryptographic operation (Signature Generation and Verification)
FCS_COP.1/Hash Cryptographic operation (Hash Algorithm)
FCS_COP.1/KeyedHash Cryptographic operation (Keyed Hash Algorithm)
FCS_RBG_EXT.1 Random Bit Generation

FCS_TLSS_EXT.1.1 The TSF shall implement [selection: *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:

- [assignment: *list of optional ciphersuites and reference to RFC in which each is defined*].

Application Note 194

The ciphersuites to be tested in the evaluated configuration are limited by this requirement.

FCS_TLSS_EXT.1.2 The TSF shall deny connections from clients requesting SSL 2.0, SSL 3.0, TLS 1.0 and [selection: *TLS 1.1, TLS 1.2, none*].

Application Note 195

All SSL versions and TLS v1.0 are denied. Any TLS versions not selected in FCS_TLSS_EXT.1.1 should be selected here. (If "none" is the selection for this element then the ST author may omit the words "and none".)

FCS_TLSS_EXT.1.3 The TSF shall [selection: *perform RSA key establishment with key size [selection: 2048 bits, 3072 bits, 4096 bits]; generate EC Diffie-Hellman parameters over NIST curves [selection: secp256r1, secp384r1, secp521r1] and no other curves; generate Diffie-Hellman parameters of size [selection: 2048 bits, 3072 bits]*].

Application Note 196

The assignments will be filled in based on the assignments performed in FCS_TLSS_EXT.1.1.

9.3 Identification and Authentication (FIA)

9.3.1 Password Management (FIA_PMG_EXT)

Family Behaviour

The TOE defines the attributes of passwords used by administrative users to ensure that strong passwords and passphrases can be chosen and maintained.

Component levelling

FIA_PMG_EXT.1 Password management requires the TSF to support passwords with varying composition requirements, minimum lengths, maximum lifetime, and similarity constraints.

Management: FIA_PMG_EXT.1

No management functions.

Audit: FIA_PMG_EXT.1

No specific audit requirements.

9.3.1.1 FIA_PMG_EXT.1 Password Management

FIA_PMG_EXT.1 Password Management

Hierarchical to: No other components.

Dependencies: No other components.

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: [selection: *“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, “)”, [assignment: other characters]*];

b) Minimum password length shall be configurable to between [*assignment: minimum number of characters supported by the TOE*] and [*assignment: number of characters greater than or equal to 15*] characters.

Application Note 203

The ST author selects the special characters that are supported by the TOE. They may optionally list additional special characters supported using the assignment. "Administrative passwords" refers to passwords used by Administrators at the local console, over protocols that support passwords, such as SSH and HTTPS, or to grant configuration data that supports other SFRs in the Security Target.

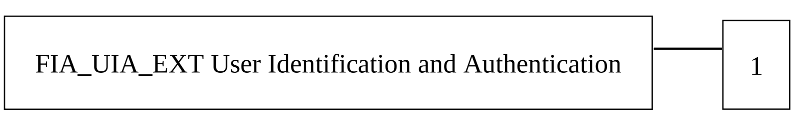
The second assignment should be configured with the largest minimum password length the Security Administrator can configure.

9.3.2 User Identification and Authentication (FIA_UIA_EXT)

Family Behaviour

The TSF allows certain specified actions before the non-TOE entity goes through the identification and authentication process.

Component levelling



FIA_UIA_EXT.1 User Identification and Authentication requires Administrators (including remote Administrators) to be identified and authenticated by the TOE, providing assurance for that end of the communication path. It also ensures that every user is identified and authenticated before the TOE performs any mediated functions

Management: FIA_UIA_EXT.1

The following actions could be considered for the management functions in FMT:

a) Ability to configure the list of TOE services available before an entity is identified and authenticated

Audit: FIA_UIA_EXT.N

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) All use of the identification and authentication mechanism
b) Provided user identity, origin of the attempt (e.g. IP address)

9.3.2.1 FIA_UIA_EXT.1 User Identification and Authentication

FIA_UIA_EXT.1 User Identification and Authentication

Hierarchical to: No other components.

Dependencies: FTA_TAB.1 Default TOE Access Banners

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;

[selection: *no other actions, automated generation of cryptographic keys, [assignment: list of services, actions performed by the TSF in response to non-TOE requests]*].

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

Application Note 204

This requirement applies to users (Administrators and external IT entities) of services available from the TOE directly, and not services available by connecting through the TOE. While it should be the case that few or no services are available to external entities prior to identification and authentication, if there are some available (perhaps ICMP echo) these should be listed in the assignment statement; if automated generation of cryptographic keys is supported without administrator authentication, the option "automated generation of cryptographic keys" should be selected; otherwise "no other actions" should be selected.

Authentication can be password-based through the local console or through a protocol that supports passwords (such as SSH), or be certificate based (such as SSH, TLS).

For communications with external IT entities (an audit server, for instance), such connections must be performed in accordance with FTP_ITC.1, whose protocols perform identification and authentication. This means that such communications (e.g., establishing the IPsec connection to the authentication server) would not have to be specified in the assignment, since establishing the connection “counts” as initiating the identification and authentication process.

According to the application note for FMT_SMR.2, for distributed TOEs at least one TOE component has to support the authentication of Security Administrators according to FIA_UIA_EXT.1 and FIA_UAU_EXT.2 but not necessarily all TOE components. In case not all TOE components support this way of authentication for Security Administrators the TSS shall describe how Security Administrators are authenticated and identified.

9.3.3 User authentication (FIA_UAU_EXT)

Family Behaviour

Provides for a locally based administrative user authentication mechanism

Component levelling

FIA_UAU_EXT.2 The password-based authentication mechanism provides administrative users a locally based authentication mechanism.

Management: FIA_UAU_EXT.2

The following actions could be considered for the management functions in FMT:

a) None

Audit: FIA_UAU_EXT.2

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) Minimal: All use of the authentication mechanism

9.3.3.1 FIA_UAU_EXT.2 Password-based Authentication Mechanism

FIA_UAU_EXT.2 Password-based Authentication Mechanism

Hierarchical to: No other components.

Dependencies: No other components.

FIA_UAU_EXT.2.1 The TSF shall provide a local password-based authentication mechanism, [selection: *[assignment: other authentication mechanism(s)], no other authentication mechanism*] to perform local administrative user authentication.

Application Note 205

The assignment should be used to identify any additional local authentication mechanisms supported. Local authentication mechanisms are defined as those that occur through the local console; remote administrative sessions (and their associated authentication mechanisms) are specified in FTP_TRP.1/Admin.

According to the application note for FMT_SMR.2, for distributed TOEs at least one TOE component has to support the authentication of Security Administrators according to FIA_UIA_EXT.1 and FIA_UAU_EXT.2 but not necessarily all TOE components. In case not all TOE components support this way of authentication for Security Administrators the TSS shall describe how Security Administrators are authenticated and identified.

9.3.4 Authentication using X.509 certificates (FIA_X509_EXT)

Family Behaviour

This family defines the behaviour, management, and use of X.509 certificates for functions to be performed by the TSF. Components in this family require validation of certificates according to a specified set of rules, use of certificates for authentication for protocols and integrity verification, and the generation of certificate requests.

Component levelling

FIA_X509_EXT.1 X509 Certificate Validation, requires the TSF to check and validate certificates in accordance with the RFCs and rules specified in the component.

FIA_X509_EXT.2 X509 Certificate Authentication, requires the TSF to use certificates to authenticate peers in protocols that support certificates, as well as for integrity verification and potentially other functions that require certificates.

FIA_X509_EXT.3 X509 Certificate Requests, requires the TSF to be able to generate Certificate Request Messages and validate responses.

Management: FIA_X509_EXT.1, FIA_X509_EXT.2, FIA_X509_EXT.3

The following actions could be considered for the management functions in FMT:
a) Remove imported X.509v3 certificates
b) Approve import and removal of X.509v3 certificates
c) Initiate certificate requests

Audit: FIA_X509_EXT.1, FIA_X509_EXT.2, FIA_X509_EXT.3

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) Minimal: No specific audit requirements are specified.

9.3.4.1 FIA_X509_EXT.1 X.509 Certificate Validation

FIA_X509_EXT.1 X.509 Certificate Validation

Hierarchical to: No other components
Dependencies: FIA_X509_EXT.2 X.509 Certificate Authentication

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

- RFC 5280 certificate validation and certification path validation.
- The certification 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 [selection: *the Online Certificate Status Protocol (OCSP) as specified in RFC 6960, a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3, Certificate Revocation List (CRL) as specified in RFC 5759 Section 5, no revocation method*]
- The TSF shall validate the extendedKeyUsage field according to the following rules: [assignment: *rules that govern contents of the extendedKeyUsage field that need to be verified*].

Application Note 206

FIA_X509_EXT.1.1 lists the rules for validating certificates. The ST author selects whether revocation status is verified using OCSP or CRLs. If the TOE is distributed and X.509 based authentication is being used to authenticate the protocol selected in FPT_ITT.1, certificate revocation checking is optional. It is optional *because there are additional requirements surrounding the enabling and disabling of the FPT_ITT channel defined in FCO_CPC_EXT.1. If revocation is not supported the ST author selects no revocation method. The ST author fills in the assignment with rules that may apply to other requirements in the ST. For instance, if a protocol such as TLS that uses certificates is specified in the ST, then certain values for the extendedKeyUsage field (e.g., "Server Authentication Purpose") could be specified.*

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

Application Note 207

This requirement applies to certificates that are used and processed by the TSF and restricts the certificates that may be added as trusted CA certificates.

9.3.4.2 FIA_X509_EXT.2 X509 Certificate Authentication

FIA_X509_EXT.2 X.509 Certificate Authentication

Hierarchical to: No other components

Dependencies: FIA_X509_EXT.1 X.509 Certificate Validation

FIA_X509_EXT.2.1 The TSF shall use X.509v3 certificates as defined by RFC 5280 to support authentication for [selection*: DTLS, HTTPS, IPsec, TLS, SSH, [assignment: other protocols], no protocols*], and [selection: *code signing for system software updates, code signing for integrity verification, [assignment: other uses], no additional uses*].

Application Note 208

If the TOE specifies the implementation of communications protocols that perform peer authentication using certificates, the ST author either selects or assigns the protocols that are specified; otherwise, they select "no protocols". Protocols that do not use X.509 based peer authentication include SSH, where ssh-rsa, ecdsa-sha2-nistp256, ecdsa-sha2-nistp384, and/or ecdsa-sha2-nistp521 are selected. The TOE may also use certificates for other purposes; the second selection and assignment are used to specify these cases.

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

Application Note 209

Often a connection must be established to check the revocation status of a certificate – either to download a CRL or to perform a lookup using OCSP. The selection is used to describe the behaviour in the event that such a connection cannot be established (for example, due to a network error). If the TOE has determined the certificate valid according to all other rules in FIA_X509_EXT.1, the behaviour indicated in the selection determines the validity. The TOE must not accept the certificate if it fails any of the other validation rules in

FIA_X509_EXT.1. If the Administrator-configured option is selected by the ST Author, the ST Author also selects the corresponding function in FMT_SMF.1.

If the TOE is distributed and FIA_X509_EXT.1/ITT is selected, then certificate revocation checking is optional. This is due to additional authorization actions being performed in the enabling and disabling of the intra-TOE trusted channel as defined in FCO_CPC_EXT.1. In this case, a connection is not required to determine certificate validity and this SFR is trivially satisfied.

9.3.4.3 FIA_X509_EXT.3 X.509 Certificate Requests

FIA_X509_EXT.3 X.509 Certificate Requests

Hierarchical to: No other components

Dependencies: FCS_CKM.1 Cryptographic Key Generation

FIA_X509_EXT.1 X.509 Certificate Validation

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

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

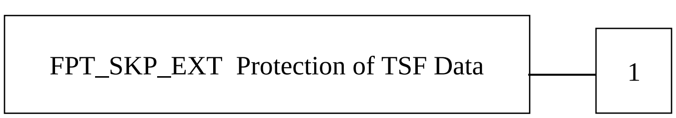
9.4 Protection of the TSF (FPT)

9.4.1 Protection of TSF Data (FPT_SKP_EXT)

Family Behaviour

Components in this family address the requirements for managing and protecting TSF data, such as cryptographic keys. This is a new family modelled after the FPT_PTD Class.

Component levelling



FPT_SKP_EXT.1 Protection of TSF Data (for reading all symmetric keys), requires preventing symmetric keys from being read by any user or subject. It is the only component of this family.

Management: FPT_SKP_EXT.1

The following actions could be considered for the management functions in FMT: a) There are no management activities foreseen.

Audit: FPT_SKP_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) There are no auditable events foreseen.

9.4.1.1 FPT_SKP_EXT.1 Protection of TSF Data (for reading of all symmetric keys)

Hierarchical to: No other components.
Dependencies: No other components.

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

Application Note 210
The intent of this requirement is for the device to protect keys, key material, and authentication credentials from unauthorized disclosure. This data should only be accessed for the purposes of their assigned security functionality, and there is no need for them to be displayed/accessed at any other time. This requirement does not prevent the device from providing indication that these exist, are in use, or are still valid. It does, however, restrict the reading of the values outright.

9.4.2 Protection of Administrator Passwords (FPT_APW_EXT.1)

Family Behaviour

Components in this family ensure that the TSF will protect plaintext credential data such as passwords from unauthorized disclosure.

Component levelling

FPT_APW_EXT.1 Protection of Administrator passwords requires that the TSF prevent plaintext credential data from being read by any user or subject.

Management: FPT_APW_EXT.1

The following actions could be considered for the management functions in FMT:

a) No management functions.

Audit: FPT_APW_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) No audit necessary.

9.4.2.1 FPT_APW_EXT.1 Protection of Administrator Passwords

Hierarchical to: No other components
Dependencies: No other components.

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

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

Application Note 211
The intent of the requirement is that raw password authentication data is not stored in the clear, and that no user or Administrator is able to read the plaintext password through "normal" interfaces. An all-powerful Administrator could directly read memory to capture a password but is trusted not to do so. Passwords should be obscured during entry on the local console in accordance with FIA_UAU.7.

A single user associated with the Security Administrator role does not necessarily have to be able to perform all security management functions defined in FMT_SMF.1 and does not necessarily have to able to perform local and remote administration. All users associated with the Security Administrator role together need to be able to perform all security management functions defined in FMT_SMF.1 (mandatory and selected ones) and need to be able to perform local and remote administration.

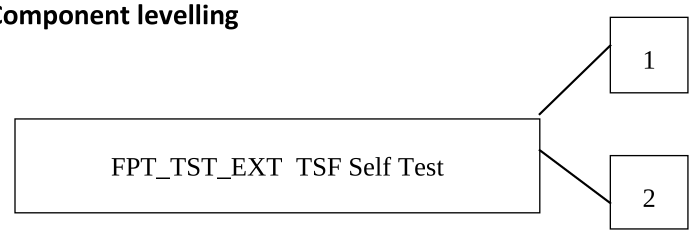
This implies that a user that can perform only a single security management function defined in FMT_SMF.1 needs to be regarded as Security Administrator of the TOE.

9.4.3 TSF Self-Test (FPT_TST_EXT)

Family Behaviour

Components in this family address the requirements for self-testing the TSF for selected correct operation.

Component levelling



FPT_TST_EXT.1 TSF Self-Test requires a suite of self-tests to be run during initial start-up in order to demonstrate correct operation of the TSF.

FPT_TST_EXT.2 Self-tests based on certificates applies when using certificates as part of self-test, and requires that the self-test fails if a certificate is invalid.

Management: FPT_TST_EXT.1, FPT_TST_EXT.2

The following actions could be considered for the management functions in FMT:

a) No management functions.

Audit: FPT_TST_EXT.1, FPT_TST_EXT.2

The following actions should be considered for audit if FAU_GEN Security audit data generation is included in the PP/ST:

a) Indication that TSF self-test was completed
b) Failure of self-test

9.4.3.1 FPT_TST_EXT.1 TSF testing

Hierarchical to: No other components.
Dependencies: No other components.

FPT_TST_EXT.1.1 The TSF shall run a suite of the following self-tests [selection: during initial start-up (on power on), periodically during normal operation, at the request of the authorised user, at the conditions [assignment: conditions under which self-tests should occur]] to demonstrate the correct operation of the TSF: [assignment: list of self-tests run by the TSF].

Application Note 212
It is expected that self-tests are carried out during initial start-up (on power on). Other options should only be used if the developer can justify why they are not carried out during initial start-up. It is expected that at least self-tests for verification of the integrity of the firmware and software as well as for the correct operation of cryptographic functions necessary to fulfil the SFRs will be performed. If not all self-tests are performed during start-up multiple iterations of this SFR are used with the appropriate options selected. In

future versions of this cPP the suite of self-tests will be required to contain at least mechanisms for measured boot including self-tests of the components which perform the measurement.

Non-distributed TOEs may internally consist of several components that contribute to enforcing SFRs. Self-testing shall cover all components that contribute to enforcing SFRs and verification of integrity shall cover all software that contributes to enforcing SFRs on all components.

For distributed TOEs all TOE components have to perform self-tests. This does not necessarily mean that each TOE component has to carry out the same self-tests: the ST describes the applicability of the selection (i.e. when self-tests are run) and the final assignment (i.e. which self-tests are carried out) to each TOE component.

Application Note 213
If certificates are used by the self-test mechanism (e.g. for verification of signatures for integrity verification), certificates are validated in accordance with FIA_X509_EXT.1 and should be selected in FIA_X509_EXT.2.1. Additionally, FPT_TST_EXT.2 must be included in the ST.

FPT_TST_EXT.2 Self-tests based on certificates
Hierarchical to: No other components.
Dependencies: No other components.

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

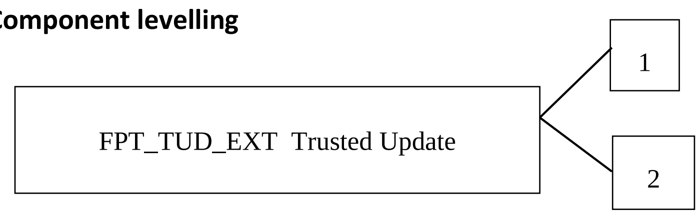
Application Note 214
Certificates may optionally be used for self-tests (FPT_TST_EXT.1.1). This element must be included in the ST if certificates are used for self-tests. If "code signing for integrity verification" is selected in FIA_X509_EXT.2.1, FPT_TST_EXT.2 must be included in the ST. Validity is determined by the certification path and the expiration date. If the self-test is executed as part of TOE initialization (e.g. boot), there is no expectation of a revocation status check as the necessary functionality, configuration, or infrastructure required to perform such check might not be available.

9.4.4 Trusted Update (FPT_TUD_EXT)

Family Behaviour

Components in this family address the requirements for updating the TOE firmware and/or software.

Component levelling



FPT_TUD_EXT.1 Trusted Update requires management tools be provided to update the TOE firmware and software, including the ability to verify the updates prior to installation.

FPT_TUD_EXT.2 Trusted update based on certificates applies when using certificates as part of trusted update and requires that the update does not install if a certificate is invalid.

Management: FPT_TUD_EXT.1

The following actions could be considered for the management functions in FMT:

a) Ability to update the TOE and to verify the updates
b) Ability to update the TOE and to verify the updates using the digital signature capability (FCS_COP.1/SigGen) and [selection: *no other functions, [assignment: other cryptographic functions (or other functions) used to support the update capability]*]
c) Ability to update the TOE, and to verify the updates using [selection: *digital signature, published hash, no other mechanism*] capability prior to installing those updates

Audit: FPT_TUD_EXT.1, FPT_TUD_EXT.2

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) Initiation of the update process.
b) Any failure to verify the integrity of the update

9.4.4.1 FPT_TUD_EXT.1 Trusted update

Hierarchical to: No other components
Dependencies: FCS_COP.1/SigGen Cryptographic operation (for Cryptographic Signature and Verification), or
FCS_COP.1/Hash Cryptographic operation (for cryptographic hashing)

FPT_TUD_EXT.1.1 The TSF shall provide [assignment: Administrators] the ability to query the currently executing version of the TOE firmware/software and [selection: the most recently installed version of the TOE firmware/software; no other TOE firmware/software version].

Application Note 215
If a trusted update can be installed on the TOE with a delayed activation the version of both the currently executing image and the installed but inactive image must be provided. In this case the option 'the most recently installed version of the TOE firmware/software' needs to be chosen from the selection in FPT_TUD_EXT.1.1 and the TSS needs to describe how and when the inactive version becomes active. If all trusted updates become active as part of the installation process, only the currently executing version needs to be provided. In this case the option 'no other TOE firmware/software version' shall be chosen from the selection in FPT_TUD_EXT.1.1..

For a distributed TOE, the method of determining the installed versions on each component of the TOE is described in the operational guidance.

FPT_TUD_EXT.1.2 The TSF shall provide [assignment: Administrators] the ability to manually initiate updates to TOE firmware/software and [selection: support automatic checking for updates, support automatic updates, no other update mechanism].

Application Note 216

The selection in FPT_TUD_EXT.1.2 distinguishes the support of automatic checking for updates and support of automatic updates. The first option refers to a TOE that checks whether a new update is available, communicates this to the Administrator (e.g. through a message during an Administrator session, through log files) but requires some action by the Administrator to actually perform the update. The second option refers to a TOE that checks for updates and automatically installs them upon availability.

The TSS explains what actions are involved in the TOE support when using the "support automatic checking for updates" or "support automatic updates" selections.

When published hash values (see FPT_TUD_EXT.1.3) are used to protect the trusted update mechanism, the TOE must not automatically download the update file(s) together with the hash value (either integrated in the update file(s) or separately) and automatically install the update without any active authorization by the Security Administrator, even when the calculated hash value matches the published hash value. When using published hash values to protect the trusted update mechanism, the option "support of automatic updates" must not be used (automated checking for updates is permitted, though). The TOE may automatically download the update file(s) themselves but not to the hash value. For the published hash approach, it is intended that a Security Administrator is always required to give active authorisation for installation of an update (as described in more detail under FPT_TUD_EXT.1.3) below. Due to this, the type of update mechanism is regarded as "manually initiated update", even if the update file(s) may be downloaded automatically. A fully automated approach (without Security Administrator intervention) can only be used when "digital signature mechanism" is selected in FPT_TUD_EXT.1.3 below.

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

Application Note 217

The digital signature mechanism referenced in the selection of FPT_TUD_EXT.1.3 is one of the algorithms specified in FCS_COP.1/SigGen. The published hash referenced in FPT_TUD_EXT.1.3 is generated by one of the functions specified in FCS_COP.1/Hash. The ST author should choose the mechanism implemented by the TOE; it is acceptable to implement both mechanisms.
When published hash values are used to secure the trusted update mechanism, an active authorization of the update process by the Security Administrator is always required. The secure transmission of an authentic hash value from the developer to the Security Administrator is one of the key factors to protect the trusted update mechanism when using published hashes and the guidance documentation needs to describe how this transfer has to be performed. For the verification of the trusted hash value by the Security Administrator different use cases are possible. The Security Administrator could obtain the published hash value as well as the update file(s) and perform the verification outside the TOE while the hashing of the update file(s) could be done by the TOE or by other means. Authentication as Security Administrator and initiation of the trusted update would in this case be regarded as "active authorization" of the trusted update. Alternatively, the Administrator could provide the TOE with the published hash value together with the update file(s) and the hashing and hash comparison is performed by the TOE. In case of successful hash verification, the TOE can perform the update without any additional step by the Security Administrator. Authentication as Security Administrator and sending the hash value to the TOE is regarded as "active authorization" of the trusted

update (in case of successful hash verification), because the Administrator is expected to load the hash value only to the TOE when intending to perform the update. As long as the transfer of the hash value to the TOE is performed by the Security Administrator, loading of the update file(s) can be performed by the Security Administrator or can be automatically downloaded by the TOE from a repository.

If the digital signature mechanism is selected, the verification of the signature shall be performed by the TOE itself. For the published hash option, the verification can be done by the TOE itself as well as by the Security Administrator. In the latter case use of TOE functionality for the verification is not mandated, so verification could be done using non-TOE functionality of the device containing the TOE or without using the device containing the TOE.

For distributed TOEs all TOE components shall support Trusted Update. The verification of the signature or hash on the update shall either be done by each TOE component itself (signature verification) or for each component (hash verification).
Updating a distributed TOE might lead to the situation where different TOE components are running different software versions. Depending on the differences between the different software versions the impact of a mixture of different software versions might be no problem at all or critical to the proper functioning of the TOE. The TSS shall detail the mechanisms that support the continuous proper functioning of the TOE during trusted update of distributed TOEs.

Application Note 218
Future versions of this cPP will mandate the use of a digital signature mechanism for trusted updates.

Application Note 219
If certificates are used by the update verification mechanism, certificates are validated in accordance with FIA_X509_EXT.1 and should be selected in FIA_X509_EXT.2.1. Additionally, FPT_TUD_EXT.2 must be included in the ST.

Application Note 220
"Update" in the context of this SFR refers to the process of replacing a non-volatile, system resident software component with another. The former is referred to as the NV image, and the latter is the update image. While the update image is typically newer than the NV image, this is not a requirement. There are legitimate cases where the system owner may want to rollback a component to an older version (e.g. when the component manufacturer releases a faulty update, or when the system relies on an undocumented feature no longer present in the update). Likewise, the owner may want to update with the same version as the NV image to recover from faulty storage.

All discrete firmware and software elements (e.g. applications, drivers, and kernel) of the TSF need to be protected, i.e. they should either be digitally signed by the corresponding manufacturer and subsequently verified by the mechanism performing the update or a hash should be published for them which needs to be verified before the update.

9.4.4.2 FPT_TUD_EXT.2 Trusted update based on certificates

Hierarchical to: No other components
Dependencies: FPT_TUD_EXT.1

FPT_TUD_EXT.2.1 The TSF shall not install an update if the code signing certificate is deemed invalid.

FPT_TUD_EXT.2.2 When the certificate is deemed invalid because the certificate has expired, the TSF shall [selection: *allow the Administrator to choose whether to accept the certificate in these cases, accept the certificate, not accept the certificate*].

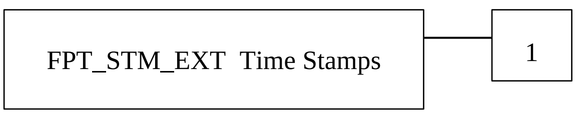
> ***Application Note 221***
> *Certificates may optionally be used for code signing of system software updates (FPT_TUD_EXT.1.3). This element must be included in the ST if certificates are used for validating updates. If "code signing for system software updates" is selected in FIA_X509_EXT.2.1, FPT_TUD_EXT.2 must be included in the ST.*
>
> *Validity is determined by the certification path, the expiration date, and the revocation status in accordance with FIA_X509_EXT.1. For expired certificates the author of the ST selects whether the certificate shall be accepted, rejected or the choice is left to the Administrator to accept or reject the certificate.*

9.4.5 Time stamps (FPT_STM_EXT)

Family Behaviour

Components in this family extend FPT_STM requirements by describing the source of time used in timestamps.

Component levelling



FPT_STM_EXT.1 Reliable Time Stamps is hierarchic to FPT_STM.1: it requires that the TSF provide reliable time stamps for TSF and identifies the source of the time used in those timestamps.

Management: FPT_STM_EXT.1

The following actions could be considered for the management functions in FMT:

a) Management of the time
b) Administrator setting of the time.

Audit: FTA_SSL_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) Discontinuous changes to the time.

9.4.5.1 **FPT_STM_EXT.1 Reliable Time Stamps**

Hierarchical to: No other components
Dependencies: No other components.

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 [selection: allow the Security Administrator to set the time, synchronise time with an NTP server].

Application Note 222

Reliable time stamps are expected to be used with other TSF, e.g. for the generation of audit data to allow the Security Administrator to investigate incidents by checking the order of events and to determine the actual local time when events occurred. The decision about the required level of accuracy of that information is up to the Administrator.

The TOE depends on time and date information, either provided by a local real-time clock that is manually managed by the Security Administrator or through the use of one or more external NTP servers. The corresponding option(s) shall be chosen from the selection in FPT_STM_EXT.1.2. The use the automatic synchronisation with an external NTP server is recommended but not mandated. Note that for the communication with an external NTP server, FCS_NTP_EXT.1 shall be claimed. The ST author describes in the TSS how the external time and date information is received by the TOE and how this information is maintained.

The term "reliable time stamps" refers to the strict use of the time and date information, that is provided, and the logging of all discontinuous changes to the time settings including information about the old and new time. With this information, the real time for all audit data can be determined. Note, that all discontinuous time changes, Administrator actuated or changed via an automated process, must be audited. No audit is needed when time is changed via use of kernel or system facilities – such as daytime (3) – that exhibit no discontinuities in time.

For distributed TOEs it is expected that the Security Administrator ensures synchronization between the time settings of different TOE components. All TOE components shall either be in sync (e.g. through synchronisation between TOE components or through synchronisation of different TOE components with external NTP servers) or the offset should be known to the Administrator for every pair of TOE components. This includes TOE components synchronized to different time zones.

9.5 TOE Access (FTA)

9.5.1 TSF-initiated Session Locking (FTA_SSL_EXT)

Family Behaviour

Components in this family address the requirements for TSF-initiated and user-initiated locking, unlocking, and termination of interactive sessions.

The extended FTA_SSL_EXT family is based on the FTA_SSL family.

Component levelling

FTA_SSL_EXT.1 TSF-initiated session locking, requires system initiated locking of an interactive session after a specified period of inactivity. It is the only component of this family.

Management: FTA_SSL_EXT.1

The following actions could be considered for the management functions in FMT:

c) Specification of the time of user inactivity after which lock-out occurs for an individual user.

Audit: FTA_SSL_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

b) Any attempts at unlocking an interactive session.

9.5.1.1 FTA_SSL_EXT.1 TSF-initiated Session Locking

Hierarchical to: No other components
Dependencies: FIA_UAU.1 Timing of authentication

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

- lock the session - disable any activity of the Administrator's data access/display devices other than unlocking the session, and requiring that the Administrator re-authenticate to the TSF prior to unlocking the session;
- terminate the session] after a Security Administrator-specified time period of inactivity.